

DATA's STORE PROGRAMBLAD

JANUAR 1985, NR. 1.

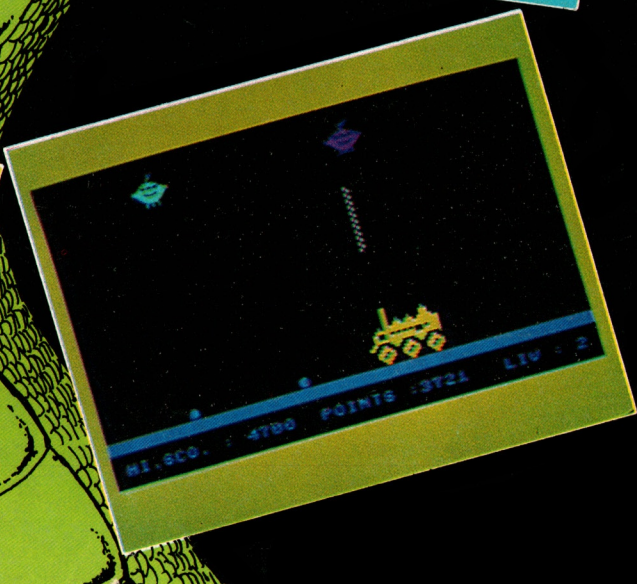
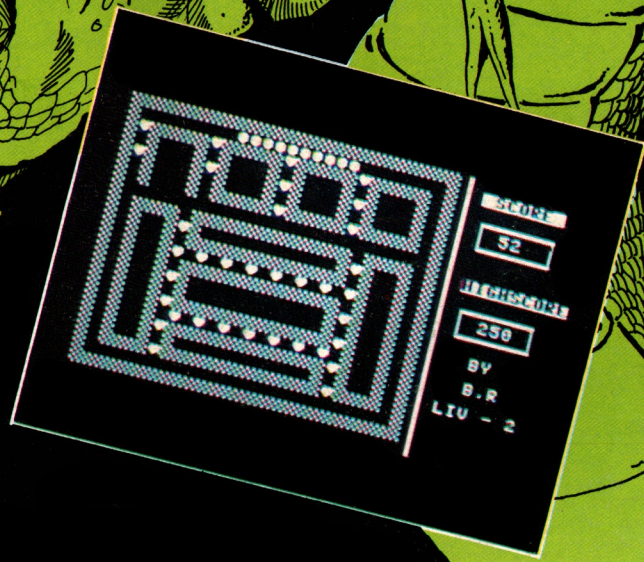
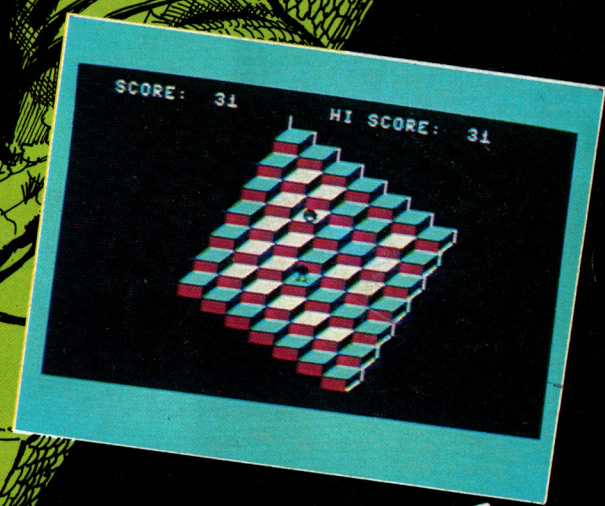
KR. 27,85.

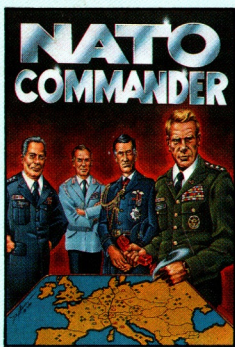
SOFT

SPECIAL

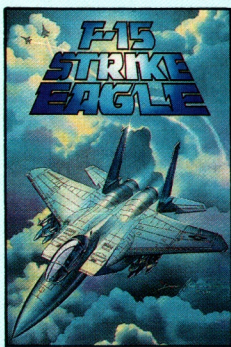
**Masser af sjove programmer til: VIC-20
Commodore 64.
Spectrum
ZX-81
BBC**

**Lambda
Amstrad
Memotech
IBM PC
Piccolo**

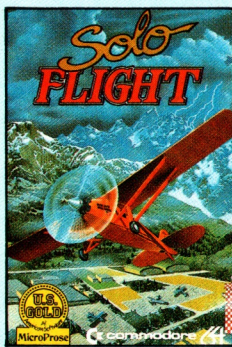




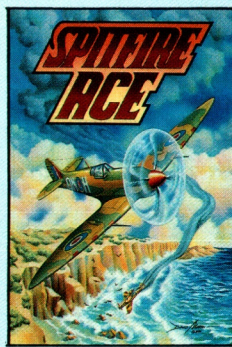
NATO COMMANDER
CBM Kass: 198,- disk: 278,-



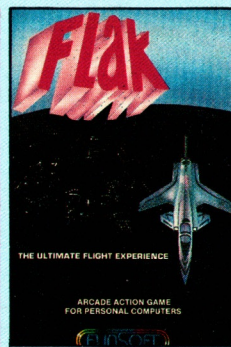
F-15 STRIKE EAGLE
SP: 198,- CBM Kass: 298,-
disk: 298,-



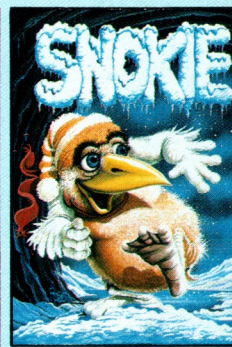
SOLO FLIGHT
SP: 198,- CBM Kass: 298,-
disk: 298,-



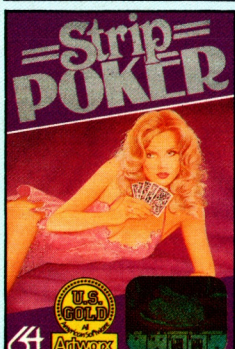
SPITFIRE ACE
CBM Kass: 198,- disk: 278,-



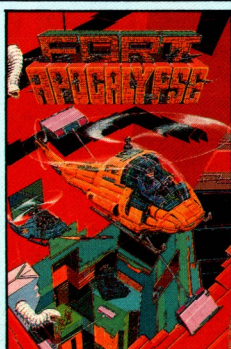
FLAK
CBM Kass: 198,- disk: 298,-



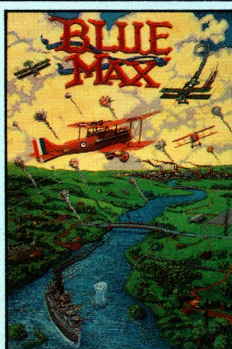
SNOKIE
CBM Kass: 198,- disk: 278,-



STRIP POKER
SP: 175,- CBM Kass: 198,-
disk: 278,-



FORT APOCALYPSE
SP: 175,- CBM Kass: 198,-
disk: 278,-



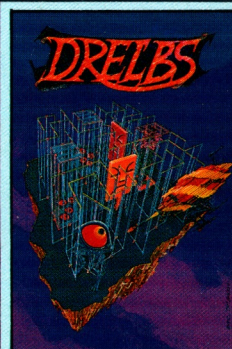
BLUE MAX
SP: 175,- CBM Kass: 198,-
disk: 278,-



ZAXXON
SP: 175,- CBM Kass: 198,-
disk: 278,-



SENTINEL
CBM Kass: 198,- disk: 278,-



DRELBS
CBM Kass: 198,- disk: 278,-

TWILIGHT APS.

Flintholm Alle 26, 2000 København F.
Tlf: 01-880734

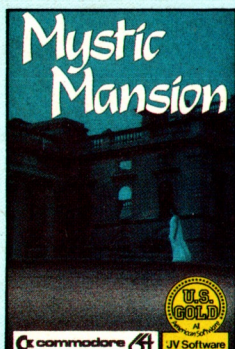
GOLD



RUSH!

NB: Kun salg til videreforhandler.

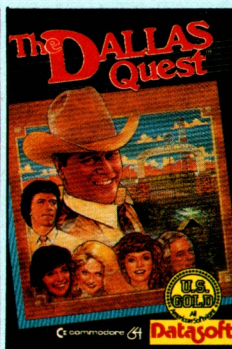
DET BEDSTE SOFTWARE FRA
DANMARKS BEDSTE DISTRIBUTØR!



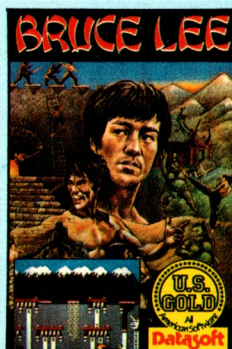
MYSTIC MANSION
CBM Kass: 198,- disk: 278,-



BEACH-HEAD
SP: 175,- CBM Kass: 198,-
disk: 278,-



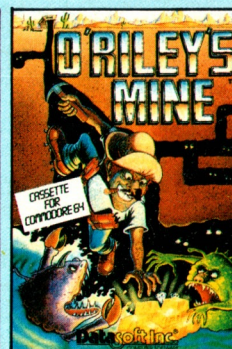
DALLAS
CBM disk: 298,-



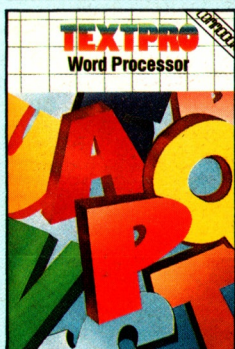
BRUCE LEE
SP: 175,- CBM Kass: 198,-
disk: 298,-



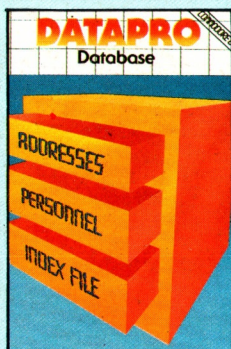
POOYAN
CBM Kass: 198,- disk: 278,-



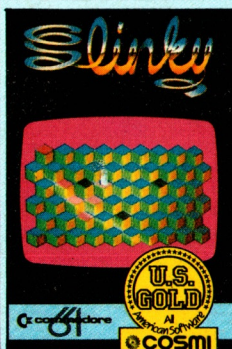
O'RILEY'S MINE
CBM Kass: 198,- disk: 278,-



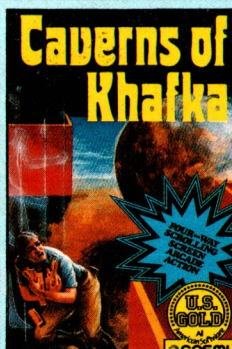
TEXTPRO
CBM Kass: 198,- disk: 278,-



DATAPRO
CBM Kass: 198,- disk: 278,-



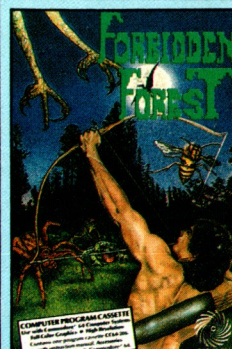
SLINKY
CBM Kass: 186,- disk: 278,-



CAVERNS OF KHAFKA
CBM Kass: 186,- disk: 278,-



AZTEC CHALLENGE
CBM Kass: 186,- disk: 278,-



FORBIDDEN FOREST
CBM Kass: 186,- disk: 278,-

SOFT SPECIAL er fremstillet af redaktionen på månedsbladet "Alt om Data". Samtlige aftrykte programlistninger er afprøvet og gengivet efter korrektur.

Redaktion:

Hans Chr. Thaysen.
Leif Bomberg
Klaus Nordfeld (ansv)
Ivan Sølvason

Udgiver:

SOFT SPECIAL udgives af Forlaget Ny Elektronik ApS, St. Kongensgade 72, 1264 København K. Giro: 9 40 60 77.

Produktion:

Hans Chr. Thaysen
ABK-Sats ApS
Partner Repro
P. Hesthavens Repro
Lassen Offset

Distribution:

Bladkompagniet

ISSN 0109-9523

Hvis du mener, du har et rigtigt godt program til "Soft Special", send da kassettebånd, disc og gerne listning til redaktionen. Alle i blade eller på anden måde offentliggjorte programmer præmieres med op til 1.000 kroner, skattefrit. Bånd mv. returneres efter brug.

Alle indsendte programmer mærkes med navn, adresse og computertype.

Afsender garanterer at programmet er originalt.

1

- | | | | |
|----|-----------------------------|----|-----------------------------|
| 4 | Søt larm på dine spil | 30 | Q-Bert / |
| 6 | Simple Logo / | | Commodore 64 |
| | Spectrum | 32 | Super-Bam / |
| 8 | Månelokomotiv / | | Commodore 64 |
| | Commodore 64 | 33 | QL Backup / QL |
| 9 | Ufo-Flight / | 34 | Stroboskop / |
| | Commodore 64 | | Commodore 64 |
| 10 | Tekstbehandling / | 34 | Digitalur / |
| | Piccolo | | Commodore 64 |
| 12 | Gunfight / Spectrum | 36 | Rundt på gulvet / |
| 13 | Turborace / Spectrum | | BBC B |
| 14 | Slimi / | 38 | Rendez-Vous / |
| | Commodore 64 | | TI 99/4A |
| 16 | Bricks / ZX 81 | 40 | Spectrum økonomi / |
| 16 | CPR-check / | | Spectrum |
| | Spectrum | 42 | Spectrum strenge / |
| 17 | Garfield II / VIC 20 | | Spectrum |
| 18 | Det gamle hus / | 43 | Multiaction / Lambda |
| | Memotech | 44 | House Defender / |
| 21 | Geometri / | | VIC 20 |
| | Commodore 64. | 45 | IBM-Days / IBM PC |
| 22 | Geografi / ZX 81 | 46 | Kanal Plus 2 / |
| 23 | Budget / | | Spectrum |
| | Commodore 64 | 47 | Happy Days / VIC 20 |
| 24 | Mastermind / | 47 | Electric Marathon / |
| | Commodore 64 | | Marathon |
| 25 | Den kinesiske mur / | 48 | Disassembler / |
| | Commodore 64 | | Amstrad |
| 25 | Skærmkoordinater / | 49 | VIC-Diagram / VIC 20 |
| | VIC 20 | 50 | Cosmic Changa / |
| 26 | Cosmic Killer / | | Commodore 64 |
| | Commodore 64 | 51 | 21 / Commodore 64 |
| 28 | Subflight / ZX 81 | 52 | Snyd (ikke) Dem selv |
| 29 | Juveler / Spectrum | | i skat! / Spectrum |

SAET

■ **Computerspil bliver først rigtig gode, når der kommer lyd på. Ville du ikke blive dødtrist til mode over et luftkampspil, hvor jagerne døde uden at "protestere". Eller racerbanen ligger hen i tavshed, som om nogen har lagt en pude over udstødningsrørene?**

Alle computere har lydkanaler indbygget. Blot programmeres de ikke på samme måde. Alligevel findes der lighedspunkter, som vi kan udnytte i en række smårutiner til Commodore 64, Spectrum og Texas TI-99/4A. Rutinerne kan nemt indbygges som underprogrammer i eksisterende spil ved hjælp af GOSUB kommandoer. Husk blot at trykke RETURN i slutningen af hver lydeffekt listning.

Småtips til lydmarkomaner

Hvis du f.eks. vil lægge lyd til en nedstyrtende flyver, er det nemmest at lave en melodifølge, som går fra en høj til en lav tone. Melodirækken skal køre så hurtigt, at man ikke hører de enkelte toner, men kun et længere piif. Dette gøres nemmest ved at lade samtlige toner indgå i en FOR...NEXT løkke.

Gælder det om at skabe et faresignal, bruger vi den modsatte melodirække. Så opnår vi lige præcis den alarmlyd, som kendes fra spændingsfilm. Det er en god idé at lade tonerækken køre igennem en FOR...NEXT løkke tre-fire gange. Så lyder det virkelig farligt.

Lyden af en kørende bil frembringes ved hjælp af sus, der køres via filtre, så vi til sidst opnår en brummelyd. Hvis vi forvrænger fire forskellige toner efter hinanden, opnår vi lyden fra et gearskifte.

Selvfølgelig kan de viste lydeffekter bruges i mange andre sammenhænge. F.eks. egner nedstyrtningsrutinen sig fint til kollisioner o.lign.

Lydeffekter til Commodore 64

COMMO
DORE
64

Lydkredsens basisadresse er 54272. Da computeren kun accepterer de to første karakterer i en variabel, forkortede vi variabelen BASIS til BA. HI er tilsvarende frekvensen HI-byte. Til gengæld kan vi godt undvære LO-byte.

REM Nedstyrt

```
10 BASIS = 54272
20 POKE BA + 24, 15
30 POKE BA + 6, 15*16
40 FOR HI = 80 TO
  10 STEP -1
50 POKE BA + 1, HI
60 POKE BA + 4, 17
70 FOR X = 0 TO 10:
```

NEXT X

80 NEXT HI

90 POKE BA + 24, 0

I linie 20 fremkommer den højeste lydstyrke. 15*16 i linie 30 sørger for en varig tone. I linie 40 fastsættes frekvenserne. Med 17 i linie 60 opstår en trekant, som kan erstattes med værdien 33. FOR...NEXT sløjfen i linie 70 sikrer den nødvendige tidsforskydning.

REM Pointtab

```
10 BASIS = 54272: Tone = 15
20 FOR HI = 60 TO
  48 STEP -2
30 POKE BA + 24, TONE
40 POKE BA + 6, 15*16
50 POKE BA + 1, HI
60 POKE BA + 4, 17
70 FOR X = 0 TO 100:
```

NEXT X

80 TONE = TONE - 1

90 NEXT HI

100 BA + 24, 0

Programmet minder om det forrige, blot reduceres lydstyrken for hvert gennemløb.

REM Alarm

```
10 BASIS = 54272
20 FOR Y = 1 TO 5
30 TONE = 10
40 FOR HI = 70 TO
  100 STEP 2
50 POKE BA + 24, TONE
60 POKE BA + 6, 15*16
70 POKE BA + 1, HI
80 POKE BA + 4, 33
90 TONE = TONE + 1
100 IF TONE 14 THEN
  TONE = 15
110 NEXT HI
120 NEXT Y
130 POKE BA + 24, 0
```

Forskellen til "nedstyrt" ligger i, at lyden høves i nivea fra 10 til 15. Med Y=1 TO 5 sikrer vi, at alarmen gennemkøres fem gange. Linie 100 er nødvendig fordi enhver lydstyrkeværdi over 15 afstedkommer en fejlmelding.

PÅ DINE SPIL

Lydeffekter
til Spectrum



REM Nedstyrt
10 FOR X = 24 TO
12 STEP -0.1
20 BEEP 0.01,x
30 NEXT X

Afspillehastigheden i linie 20 kan ændres efter behov.

REM Alarm
10 FOR y = 0 TO 2
20 FOR x = 0 TO 12
30 BEEP 0.02,x
40 NEXT x
50 NEXT y

En hel toneskala afspilles i halvtoneskridt. Tidskonstanten 0.02 kan igen ændres efter behov. Ved hjælp af FOR...NEXT sløjferne i linie 10 og 50 gentages forløbet 3 gange.

REM Pointtab
10 FOR x = 12 TO 1 STEP -1
20 BEEP 0.05,x
30 NEXT x

Værdierne X=12 TO 1 fremkommer toneskalaens C til Cis, afspillet baglæns.

REM Sejrsmelodi
10 FOR y = 0 TO 4
20 READ v,t
30 BEEP v,t
40 NEXT y
50 DATA 0.1,7,0.1,4,0.1,4
60 DATA 0.5,9,0.6,7

I denne listning gennemspilles en lille melodi bestående af tonerne G,E,F og A. Forkortelsen v,t står for "varig tone".

Lydeffekter til
Texas TI-99/4A



På Texas-computeren kaldes tonerne frem ved deres frekvenser. Bemærk i det følgende program linie 10, hvor frekvenserne fra 550 til 250 Hz afspilles med 50Hz spring. Med CALL SOUND afspilles tonen X med lydstyrken 0 og varigheden 1.

REM Pointtab
10 FOT X = 550 TO
250 STEP -50
20 CALL SOUND (1,X,0)
30 NEXT X

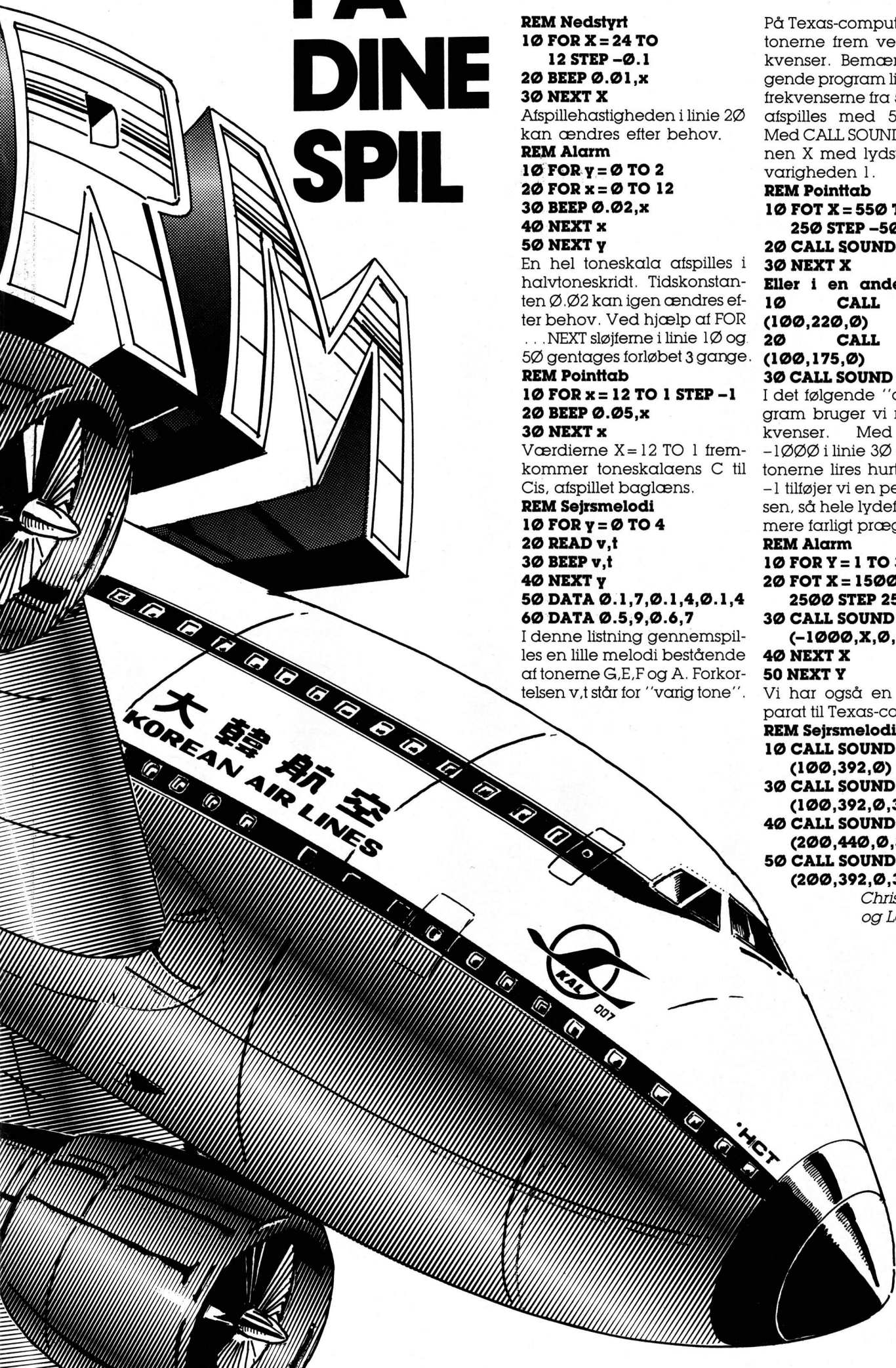
Eller i en anden version:
10 CALL SOUND
(100,220,0)
20 CALL SOUND
(100,175,0)

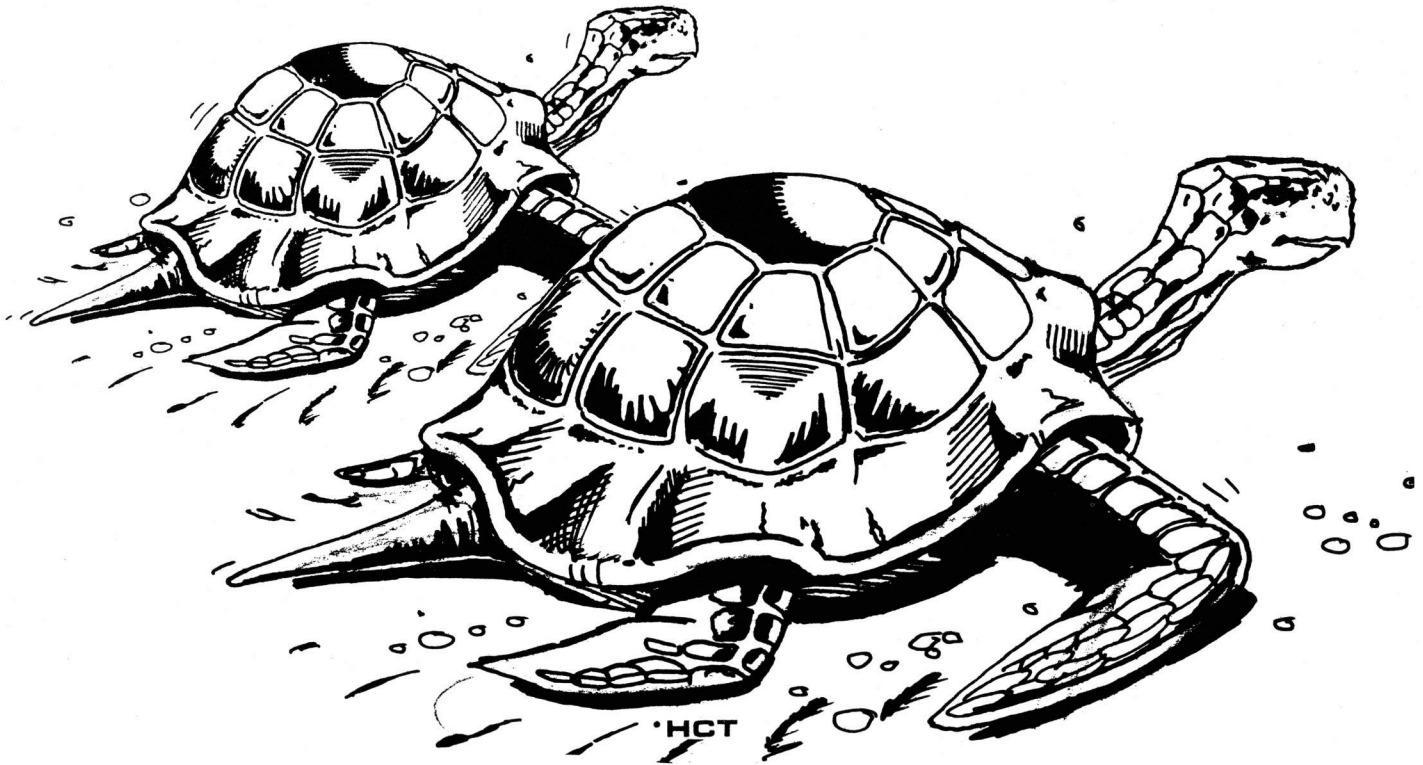
30 CALL SOUND (200,-3,0)
I det følgende "alarm" program bruger vi ret høje frekvenser. Med værdien -1000 i linie 30 opnår vi, at tonerne lires hurtigt af. Med -1 tilføjer vi en periodevis susen, så hele lydeffekten får et mere farligt præg.

REM Alarm
10 FOR Y = 1 TO 3
20 FOT X = 1500 TO
2500 STEP 250
30 CALL SOUND
(-1000,X,0,-1,0)

40 NEXT X
50 NEXT Y
Vi har også en sejrsmelodi parat til Texas-computeren:
REM Sejrsmelodi
10 CALL SOUND
(100,392,0)
30 CALL SOUND
(100,392,0,330,0)
40 CALL SOUND
(200,440,0,392,0)
50 CALL SOUND
(200,392,0,330,0).□

Christian Jensen
og Leif Bomberg





Simple Logo er en (langsom) BASIC version af sproget LOGO, men kan udmærket bruges til at lære LOGO med. De forskellige kommandoer (19 ialt) ligger i ordforrådet fra starten, og det er muligt at læse datamaten nye kommandoer. Paranteser er valgfrie. Trykkes (ENTER) i kommandofasen vises alle nylærte ord.

Simple Logo



```

10 REM // SIMPLE LOGO (C) MAZ SPORK 25.09.83 //
20 CLEAR 59299
30 LET TEGN=1: LET WORD=0: LET PROG=0: DIM W$(32,32,16): DIM Q$(32,16)
40 LET CALL=0: DIM C(32,4): LET MX=-1: LET MY=-1
50 DIM S$(16): LET X=128: LET Y=88: LET DG=0: DIM C$(16): DIM B$(16)
60 RESTORE 1060: POKE 23658,8: POKE 23562,1: POKE 23609,16
70 FOR F=59300 TO 59323: READ A: POKE F,A: NEXT F
80 CLS: LET PUT=59312: LET GET=59300: LET L=USR PUT
90 GO SUB 270
100 INPUT " "; LINE A$: IF A$="" THEN GO TO 200
110 LET C$=A$: LET ERR=0: GO SUB 120: GO TO 100
120 RESTORE 1020: LET LINE=0: FOR F=1 TO 19: READ E$,E
130 IF C$ TO LEN E$)=E$ THEN LET LINE=E: LET F=20
140 NEXT F: IF CALL=0 AND LINE=0 THEN LET ERR=1: RETURN
150 IF LINE=0 THEN RETURN
160 IF LINE=0 THEN GO SUB LINE: GO TO 200
170 LET G=0: FOR F=1 TO WORD: IF C$(G*(F)) THEN LET G=F: LET F=WORD
180 NEXT F: IF G=0 THEN LET ERR=2: GO TO 200
190 LET PROG=G: GO SUB 0920: RETURN
200 IF ERR=1 THEN PRINT #0;"1 Ikke her, 0:1"
210 IF ERR=2 THEN PRINT #0;"2 Udefineret ord, 0:1"
220 IF ERR=3 THEN PRINT #0;"3 Intet program, 0:1"
230 IF ERR=4 THEN PRINT #0;"4 Snor ikke sat, 0:1"
240 IF ERR=5 THEN PRINT #0;"5 Er defineret, 0:1"
250 IF ERR=0 THEN PAUSE 10: PAUSE 0
260 LET ERR=0: RETURN
270 POKE 23677,X: POKE 23678,Y: RETURN
280 CLS: FOR F=1 TO WORD: PRINT Q$(F): NEXT F: GO TO 100
290 FOR F=1 TO WORD
300 GO SUB 270: LET L=USR GET: LET FD=VAL C$(5 TO )
310 LET DX=INT (FD*(SIN (DG*(PI/180))))+0.5)
320 LET DY=INT (FD*(COS (DG*(PI/180))))+0.5)
330 LET X=X+DX: LET Y=Y+DY: IF TEGN THEN DRAW DX,DY
340 GO SUB 270: LET L=USR PUT: RETURN
350 GO SUB 270: LET L=USR GET: LET FD=VAL C$(4 TO )
360 LET DX=-INT (FD*(SIN (DG*(PI/180))))+0.5)
370 LET DY=-INT (FD*(COS (DG*(PI/180))))+0.5)
380 LET X=X+DX: LET Y=Y+DY: IF TEGN THEN DRAW DX,DY
390 GO SUB 270: LET L=USR PUT: RETURN
400 LET TEGN=0: RETURN
410 LET TEGN=1: RETURN
420 LET DG=0: RETURN
430 LET X=128: LET Y=88: LET DG=0: LET TEGN=1: LET MX=-1: LET MY=-1: CLS
440 LET L=USR PUT: RETURN
450 IF WORD=0 THEN LET ERR=3: RETURN
460 INPUT " "; LINE C$: LET G=0: FOR F=1 TO WORD
470 IF Q$(F)=C$ THEN LET G=F: LET F=WORD
480 NEXT F: IF G=0 THEN LET ERR=2: RETURN
490 CLS: LET LEV=1: LET F=1: PRINT Q$(G)
500 PRINT S$( TO LEV);W$(G,F)
510 IF W$(G,F, TO 6)="GENTAG" THEN LET LEV=LEV+1
520 IF W$(G,F+1, TO 6)="HERTIL" OR W$(G,F+1, TO 4)="SLUT" THEN LET LEV=LEV-1
530 IF F/20=INT (F/20) THEN INPUT "TAST <ENTER> "; LINE A$: CLS
540 IF W$(G,F, TO 4)="SLUT" THEN RETURN

```

```

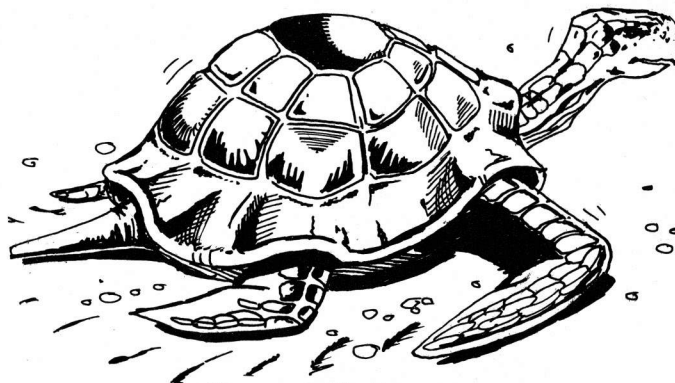
550 LET F=F+1: GO TO 0500
560 LET DG=DG+VAL C$(6 TO )-360*(DG+VAL C$(6 TO ))/359: RETURN
570 LET DG=DG-VAL C$(8 TO )+360*(DG-VAL C$(8 TO ))/359: RETURN
580 LET MX=X: LET MY=Y: RETURN
590 LET L=USR GET: IF MX=-1 THEN LET ERR=4: RETURN
600 IF TEGN THEN PLOT MX,MY: DRAW X-PEEK 23677,Y-PEEK 23678: LET L=USR PUT
610 GO SUB 270: RETURN
620 LET L=USR GET: RETURN
630 LET L=USR GET: COPY: RETURN
640 LET X=128: LET Y=88: GO SUB 270: LET TEGN=1: LET MX=-1: LET MY=-1
650 DIM C(32,4): LET CALL=0: DIM W$(32,32,16): DIM Q$(32,16): LET PROG=0
660 LET DG=0: INPUT "TAPE/MICRODRIVE? "; LINE C$: IF C$(1)="M" THEN GO TO 0690
670 LOAD "LOGO.OBJ" DATA W$( ): LOAD "LOGO.TXT" DATA Q$( )
680 DIM P(1): LOAD "LOGO.D" DATA P( ): LET WORD=P(1): CLS: RETURN
690 LOAD *"M";1;"LOGO1.D" DATA W$( ): LOAD *"M";1;"LOGO2.D" DATA Q$( )
700 DIM P(1): LOAD *"M";1;"LOGO3.D" DATA P( ): LET WORD=P(1): CLS: RETURN
710 INPUT "TAPE/MICRODRIVE? "; LINE C$
720 IF C$(1)="M" THEN GO TO 0750
730 SAVE "LOGO.OBJ" DATA W$( ): SAVE "LOGO.TXT" DATA Q$( ): DIM P(1)
740 LET P(1)=WORD: SAVE "LOGO.HEX" DATA P( ): RETURN
750 SAVE *"M";1;"LOGO1.D" DATA W$( ): SAVE *"M";1;"LOGO2.D" DATA Q$( ): DIM P(1)
760 LET P(1)=WORD: SAVE *"M";1;"LOGO3.D" DATA P( ): RETURN
770 CLS: INPUT " "; LINE C$: FOR F=1 TO WORD
780 IF C$(F) THEN LET ERR=5: RETURN
790 NEXT F: RESTORE 1020: FOR F=1 TO 19: READ D$,D
800 IF D$=C$( TO LEN D$) THEN LET ERR=5: RETURN
810 NEXT F: LET WORD=WORD+1: LET Q$(WORD)=C$: PRINT Q$(WORD)
820 LET LINE=1: LET LEV=1
830 INPUT " "; LINE C$: LET G=0: FOR F=1 TO WORD-1: IF C$(F) THEN LET G=F+1
840 NEXT F: RESTORE 1020: FOR F=1 TO 19: READ D$,D
850 IF D$=C$( TO LEN D$) THEN LET G=1
860 NEXT F: IF G=0 THEN LET ERR=2: GO SUB 200: GO TO 0830
870 IF C$( TO 4)="SLUT" OR C$( TO 6)="HERTIL" THEN LET LEV=LEV-1
880 PRINT S$( TO LEV);C$: LET W$(WORD,LINE)=C$: LET LINE=LINE+1
890 IF C$( TO 6)="GENTAG" THEN LET LEV=LEV+1
900 IF C$( TO 4)="SLUT" THEN RETURN
910 GO TO 0830
920 LET CALL=CALL+1: LET Z=0: LET LC=0: LET LOOP=0
930 LET LOOP=LOOP+1: LET C$=W$(PROG,LOOP)
940 LET C$(CALL,1)=LOOP: LET C$(CALL,2)=PROG: LET C$(CALL,3)=2: LET C$(CALL,4)=LC
950 GO SUB 120
960 LET LOOP=C$(CALL,1): LET PROG=C$(CALL,2): LET Z=C$(CALL,3): LET LC=C$(CALL,4)
970 IF LINE=-1 THEN LET Z=LOOP: LET LC=VAL C$(7 TO )-1: GO TO 0930
980 IF LINE=2 AND LC=0 THEN LET Z=0: GO TO 0930
990 IF LINE=-2 AND LC=0 THEN LET LC=LC-1: LET LOOP=2: GO TO 0930
1000 IF LINE=-3 THEN LET CALL=CALL-1: LET LINE=0: RETURN
1010 GO TO 0930
1020 DATA "KOP1",0630,"FREM",0300,"BAK",0350,"FLYV",0400
1030 DATA "HENT",0640,"GEM",0710,"HUSK",0770,"GENTAG",-1,"HERTIL",-2,"SLUT",-3
1040 DATA "KRAVL",0410,"NORD",0420,"RENS",0430,"LIGT",0450,"HØJRE",0560
1050 DATA "VENSTRE",0570,"SNOR",0580,"STRAM",0590,"TEGNING",0620
1060 DATA 33,255,231,17,0,64,1,0,24,237,176,201
1070 DATA 33,0,64,17,255,231,1,0,24,237,176,201

```

A. Foruddefinerede instruktioner

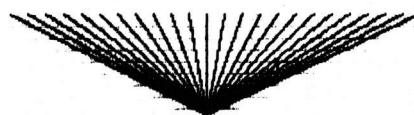
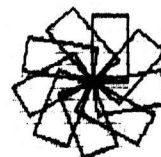
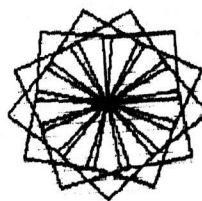
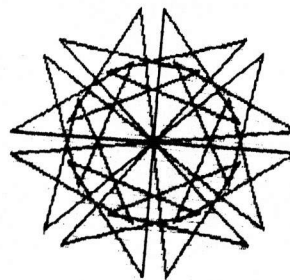
1. **FREM**
Kommandoen FREM(n) bruges til at flytte "Skildpadden"(n). Punkter eller Pixels i midlertidig retning.
- Se BAK(n).
2. **BAK**
Bruges som FREM(n) til at flytte skildpadden n Pixels i modsat retning.
3. **FLYV**
FLYV bevirker, at skildpadden ikke sætter spor, når den bevæger sig.
- Se KRAVL.
4. **KRAVL**
KRAVL får skildpadden til, modsat FLYV, at sætte spor under flytning.
5. **HENT**
Alle nulcæerte kommandoer hentes fra bånd eller Microdrive.
- Se GEM.
6. **GEM**
Gemmer alle nye kommandoer på bånd eller Microdrive.
7. **HUSK**
Dette er kommandoen til at lære computeren en ny kommando. Skriv HUSK (enter) VIFTE (enter). Computeren vil da skrive navnet VIFTE øverst på skærmen. Dernæst kan skrives kommandoer (både originale og nye) ind, som bliver lagret og skrevet. Der sluttes af med SLUT. Tre kommandoer kan kun bruges i indlæringsfasen. De er: SLUT, GENTAG(n) og HERTIL. Se disse.
8. **SLUT**
I indlæringsfasen bevirker SLUT at computeren returnerer til kommandofasen.
9. **GENTAG**
GENTAG(n) vil gentage de efterfølgende kommandoer indtil HERTIL instruktionen n gange. Bemærk at GENTAG kun kan bruges i indlæringsfasen.
- Se HERTIL.
10. **HERTIL**
Hopper tilbage til GENTAG-kommandoen, indtil dette er gjort de antal gange som beskrevet under GENTAG. Derefter fortsættes med næste kommando.

11. **NORD**
Får skildpadden til at pege mod nord (opad).
12. **RENS**
Sletter skærmen, flytter skildpadden til midten af skærmen, fjerner SNORen hvis den er sat og udfører KRAVL-instruktion.
13. **KOPI**
Kopierer skærbilledet til ZX PRINTER.
13. **TEGNING**
Viser tegningen.
15. **SNOR**
Får skildpadden til at hamre et søm i, hvor den står, for dernæst at binde en snor om sømnet.
- Se STRAM.
16. **STRAM**
Får skildpadden til at stramme snoren. Dette bevirker, at en streng bliver tegnet mellem sømnet (skildpaddens placering ved SNOR-instruktionen) og den nuværende placering. Snoren sidder stadig på sømnet.
17. **LIST**
Skriv LIST (enter) og så navnet på en nylært instruktion. Den vil da blive udskrevet på skærmen.
18. **HØJRE**
(Skrives HØJRE med ø'et som et nul!). HØJRE(n) får skildpadden til at dreje n grader mod højre.
- Se VENSTRE.
19. **VENSTRE**
VENSTRE(n) får skildpadden til at dreje n grader mod venstre.



Skærbilleder

Her er nogle billeder fra SIMPLE LOGO:



B. Fejlkode

Nr. Fejltekst	Forklaring
1. Ikke her.	En af instruktionerne GENTAG, HERTIL eller SLUT er blevet udført i normal kommandofase.
2. Udefineret ord.	Kommandoen er ikke defineret.
3. Lager tomt.	Der er forsøgt LIST i et tomt lager.
4. Snor ikke sat.	Kommandoen STRAM er udført uden først at have udført SNOR eller måske er RENS udført i mellemtiden.
5. Er defineret.	Navnet findes i forvejen.

C. Progameksemples

VIFTE	FIRKANT	BLOMST
RENS	GENTAG(4)	RENS
RENS	FREM(40)	GENTAG(13)
GENTAG(10)	HØJRE(90)	FIRKANT
FLYV	HERTIL	HØJRE(360/13)
FREM(16)	SLUT	HERTIL
HØJRE(360/13)		SLUT
KRAVL		
STRAM		
HERTIL		
SLUT		

D. Indtastning af et nyt ord

Vi vil lære maskinen det ovenstående ord VIFTE. Skriv følgende: (efter hver linie trykkes ENTER!).

HUSK
VIFTE
RENS
SNOR
GENTAG 10
FLYV
FREM 16
HØJRE 360/10
KRAVL
STRAM
HERTIL
SLUT

Nu har maskinen lært ordet VIFTE. Prøv at skrive!
VIFTE.



Behandling

```

0010 ENABLE fej1
0020 // Copyright (c) by Lars Christensen, Hillerød. Version 1.
0030 lin:=201*80; pos:=1; point:=1
0040 DIM text$ OF lin, filnavn$ OF 15, tegn$ OF 1, find$ OF 80, i$ OF 1, tom$ OF 80
0050 text$(1:lin):="" ; SCREEN$:="" ; i$:=CHR$(144) ; tom$(1:80):=""
0060 CLOSE
0070 MARGIN 0
0080 ZONE 0
0090 OPEN FILE 5, "keyboard", READ
0100 REPEAT
0110 x:=(pos-1) MOD 80+1; y:=(pos-1) DIV 80+1
0120 REPEAT
0130 IF pos(1 OR pos)2000 THEN EXEC scroll
0140 PRINT AT(x,y)
0150 tegn$:=GET$(5,1); ok!="" "(=tegn$ AND tegn$(="å"
0160 IF ok THEN SCREEN$(pos):=tegn$; pos:=pos+1; x:=x+1-80*(x=80); y:=y+1*(
x=1)
0170 UNTIL NOT ok
0180 CASE ORD(tegn$) OF
0190 WHEN 26
0200 pos:=pos-80
0210 WHEN 10
0220 pos:=pos+80
0230 WHEN 8
0240 pos:=pos-1
0250 WHEN 24
0260 pos:=pos+1
0270 WHEN 20
0280 pos:=pos+10
0290 WHEN 13
0300 pos:=((pos-1) DIV 80)*80+81
0310 WHEN 30
0320 SCREEN$(pos:((pos-1) DIV 80)*80+80):=""
0330 WHEN 31
0340 SCREEN$(pos:2000):=""
0350 WHEN 9
0360 x:=(pos-1) DIV 80+1)*80; SCREEN$(pos:x):="" +SCREEN$(pos:x)
0370 WHEN 5
0380 x:=(pos-1) DIV 80+1)*80; SCREEN$(pos:x):=SCREEN$(pos+1*(pos(x):x)
0390 WHEN 127
0400 pos:=pos-1
0410 IF pos(1 THEN EXEC scroll
0420 SCREEN$(pos):=""
0430 OTHERWISE
0440 EXEC special
0450 ENDCASE
0460 UNTIL FALSE
0470 PROC special
0480 text$(point:point+1999):=SCREEN$
0490 PRINT AT(1,1);CHR$(30);AT(79,1);CHR$(27)+CHR$(128);
0500 PRINT AT(1,1);CHR$(27);
0510 CASE ORD(tegn$) OF
0520 WHEN 27
0530 PRINT i$;" Skal programmet stoppes (j/n) ? ";
0540 IF GET$(5,1) IN "Jj" THEN STOP
0550 WHEN 1
0560 IF pos=1 AND point=1 THEN
0570 EXEC flyt(max)
0580 ELSE
0590 EXEC flyt(1)
0600 ENDIF
0610 WHEN 28
0620 IF pos+point(=lin THEN
0630 text$(pos+point:lin)=text$(pos+point-1:lin); text$(pos+point-1):=""
0640 ENDIF
0650 WHEN 29
0660 IF pos+point(=lin THEN
0670 text$(pos+point-1:lin)=text$(pos+point:lin)
0680 ENDIF
0690 WHEN 19
0700 PRINT i$;" Linie : ",(point+pos-1) DIV 80+1,
0710 PRINT ", ud af ",lin/80,". Tryk en tast.":GET$(5,1)

```

```

0720 WHEN 12
0730 PRINT i$;" Vil du slette al tekst på skærmen (J/N) ? ";
0740 IF GET$(5,1) IN "Jj" THEN text$(point:point+1999):="" ; pos:=1
0750 WHEN 6
0760 PRINT i$;" Find : "
0770 PRINT find$;CHR$(30)
0780 INPUT AT(1,2): find$
0790 x:=find$ IN text$(pos:point:lin)
0800 IF x=0 THEN
0810 PRINT CHR$(7);
0820 ELSE
0830 EXEC flyt(x+pos+point-1)
0840 ENDIF
0850 WHEN 23
0860 PRINT i$;CHR$(12)
0870 MOUNT 1
0880 DIR
0890 PRINT
0900 PRINT "Write."
0910 PRINT "Indtast tekstnavn : ";
0920 INPUT filnavn$
0930 IF filnavn$(0)="" THEN
0940 DELETE filnavn$
0950 OPEN FILE 1,filnavn$, WRITE
0960 PRINT FILE 1: text$(1:max+79);
0970 CLOSE FILE 1
0980 ENDIF
0990 WHEN 18
1000 MOUNT 1
1010 PRINT i$;CHR$(12)
1020 DIR
1030 PRINT
1040 PRINT "Read."
1050 PRINT "Indtast tekstnavn : ";
1060 INPUT filnavn$
1070 OPEN FILE 1,filnavn$, READ
1080 pos:=1; point:=1
1090 FOR xi=1 TO lin/80 DO
1100 IF EOF(1) THEN
1110 text$(x*80-79:lin):="" ; xi:=lin
1120 ELSE
1130 text$(x*80-79:x*80):=GET$(1,80)
1140 ENDIF
1150 NEXT x
1160 CLOSE FILE 1
1170 WHEN 16,3
1180 PRINT i$;
1190 offline:=FALSE; y:=point+((pos-1) DIV 80)*80; til:=max+79
1200 PRINT " Printudskriften afbrydes med et tryk på en tast."
1210 SELECT OUTPUT "printer"
1220 IF NOT offline THEN
1230 IF tegn$=CHR$(16) THEN
1240 FOR xi=y TO til STEP 80 DO
1250 PRINT text$(xi:x+79)
1260 IF KEY$(0)CHR$(0) THEN xi:=lin
1270 IF ((y-x) DIV 80-1) MOD 64=0 THEN
1280 FOR til:=1 TO 8 DO PRINT
1290 ENDIF
1300 NEXT x
1310 ELSE
1320 FOR xi=point TO point+1920 STEP 80 DO
1330 PRINT text$(xi:x+79)
1340 IF KEY$(0)CHR$(0) THEN xi:=lin
1350 NEXT x
1360 ENDIF
1370 SELECT OUTPUT "console"
1380 ELSE
1390 SELECT OUTPUT "console"
1400 PRINT CHR$(7);
1410 ENDIF
1420 WHEN 25

```



```

1430 EXEC flyt(pos+point-961)
1440 WHEN 22
1450 EXEC flyt(pos+point+959)
1460 WHEN 2
1470 PRINT i$;"Hvormange linier skal indsættes/(slettes(-)) ? "
1480 REPEAT
1490 badinput:=FALSE
1500 INPUT AT(49,1): y
1510 UNTIL NOT badinput
1520 x:=pos+point-1; y:=y*80
1530 IF y(0) THEN
1540 IF y(0) THEN
1550 IF x-y(lin) THEN
1560 text$(x:lin):=text$(x-y:lin)
1570 ELSE
1580 PRINT CHR$(7);
1590 ENDIF
1600 ELSE
1610 IF x(lin-80 AND x+y(lin) THEN
1620 text$(x+y:lin):=text$(x:lin); text$(x:x+y-1):=""
1630 ELSE
1640 PRINT CHR$(7);
1650 ENDIF
1660 ENDIF
1670 ENDIF
1680 WHEN 4 // d
1690 y:=pos+point-1; x:=(y-1) DIV 80)*80+81
1700 WHILE text$(y:1)="" AND y(x DO y:=y+1
1710 text$(x:lin):=text$(y:lin); pos:=pos+80
1720 IF y(x THEN text$(y:x-1):=""
1730 IF text$(y-1)() THEN text$(y:1):=""
1740 OTHERWISE
1750 ENDCASE
1760 SCREEN$:=text$(point:point+1999)
1770 ENDPROC special
1780 PROC scroll
1790 IF pos>2000 THEN
1800 pos:=pos-80
1810 IF point(lin-1999) THEN
1820 text$(point:point+79):=SCREEN$; SCREEN$(1:1920):=SCREEN$(81:2000)
1830 SCREEN$(1921:2000):=text$(point+2000:point+2079); point:=point+80
1840 ENDIF
1850 ELSE
1860 pos:=pos+80
1870 IF point(1) THEN
1880 point:=point-80; text$(point+2000:point+2079):=SCREEN$(1921:2000)
1890 SCREEN$(81:2000):=SCREEN$(1:1920); SCREEN$(1:80):=text$(point:point+79)
1900 ENDIF
1910 ENDIF
1920 x:=(pos-1) MOD 80+1; y:=(pos-1) DIV 80+1
1930 ENDPROC scroll
1940 PROC flyt(til)
1950 IF til(1) THEN til:=1
1960 IF til(lin) THEN til:=lin-79
1970 point:=(til-880) DIV 80)*80+1
1980 IF point(1) THEN point:=1
1990 IF point>lin-1999 THEN point:=lin-1999
2000 pos:=til-point+1
2010 ENDPROC flyt
2020 PROC fejl HANDLER
2030 CASE ERR OF
2040 WHEN 100
2050 badinput:=TRUE; filnavn$=""
2060 CONTINUE
2070 WHEN 211
2080 offline:=TRUE
2090 CONTINUE
2100 WHEN 107,118
2110 badinput:=TRUE
2120 CONTINUE
2130 WHEN 208
2140 MOUNT 1
2150 RETRY
2160 WHEN 214
2170 PRINT
2180 IF filnavn$() THEN PRINT "Findes ikke ! Tryk på en tast.";GET$(5,1)
2190 SCREEN$:=text$(point:point+1999)
2200 RETURN
2210 OTHERWISE
2220 PRINT CHR$(12);"Fatal fejl opstået. Programmet stoppet."
2230 PRINT "Fejlen har nummer ",ERR," ";
2240 IF ERR=200 THEN PRINT "Diskette eller Printer-fejl !"
2250 PRINT " og er opstået i linie ",SYS(2),"."
2260 STOP
2270 CONTINUE
2280 ENDCASE
2290 ENDPROC fejl
2300 FUNC max
2310 x:=lin-79
2320 WHILE text$(x:x+79)=tom$ AND x)1 DO x:=x-80
2330 RETURN x
2340 ENDFUNC max

```

■ Skriver du tit officielle breve? Laver du tit skriftlige opgaver, hvor det kræves at der afleveres maskinskrevet? Eller kan du bare godt lide at dine opgaver bliver afleveret på en pæn måde?

Så er dette Piccoloprogram det rigtige til dig, med en tekstbehandling kan du på en let måde skrive dine tekster med en kvalitet, der er højere end den, man normalt opnår med en skrivemaskine.

Under selve indtastningen skriver du bare løs uden de store tanker om, hvordan det egentlig kommer til at se ud. Når det er gjort kan du i ro og mag sidde og rette på teksten.

Du kan f.eks. lave pæne afsnit og en lige højre margin. Mens du på en almindelig skrivemaskine brugte fjumrevæske i spandevise, retter du, med et tekstbehandlingsprogram, dine fejl lige så let som du lavede dem. Dette viste program kan køre på en PICCOLO med RcComal80 (CP/M).

Som før nævnt skrives al tekst ved bare at skrive løs. Ud over det kan alle PICCOLO'ens redigerings-taster anvendes. Det vil sige cursorpilene, return, rubout, insert, delete samt clear.

Desuden findes der en del andre funktioner, som aktiveres ved at holde CTRL-tasten nede samtidig med at man trykker på den tast der vælger funktionen. Disse er følgende:

↑ sletter teksten linien ud.
rubout sletter teksten fra cursorpositionen og resten af skærmen.

A anbringer cursoren allerforrest i teksten. Hvis den allerede stod der, vil den blive anbragt tilsidst i teksten.

S (status) fortæller hvilken linie cursoren står på, samt hvormange linier der er ialt.

F (find) bruges til at finde et bestemt ord. Man indtaster det søgte ord, og cursoren vil blive flyttet til den første forekomst af ordet. Hvis det ikke findes får man et beep i stedet for.

W(write) gemmer teksten på disketten til senere brug.

Man skal bare indtaste det ønskede navn.

R (read) læser en tekst fra disketten.

V flytter cursoren 12 linier længere frem i teksten.

Y flytter den tilsvarende tilbage.

B bruges til at indsætte eller slette hele linier. Man indtaster antallet af linier; positivt tal hvis man vil indsætte, negativt hvis man vil slette.

P (printer) udskriver teksten fra cursorpositionen til slutningen på printeren. Man kan stoppe udskriften i utide ved at trykke på en tast. Hvis printeren ikke er klar får man et beep.

C (copy) kopierer skærmbilledet over på printeren.

Ø indsætter et mellemrum i hele teksten. (Som insert).

Å sletter en karakter i hele teksten. (Som delete).

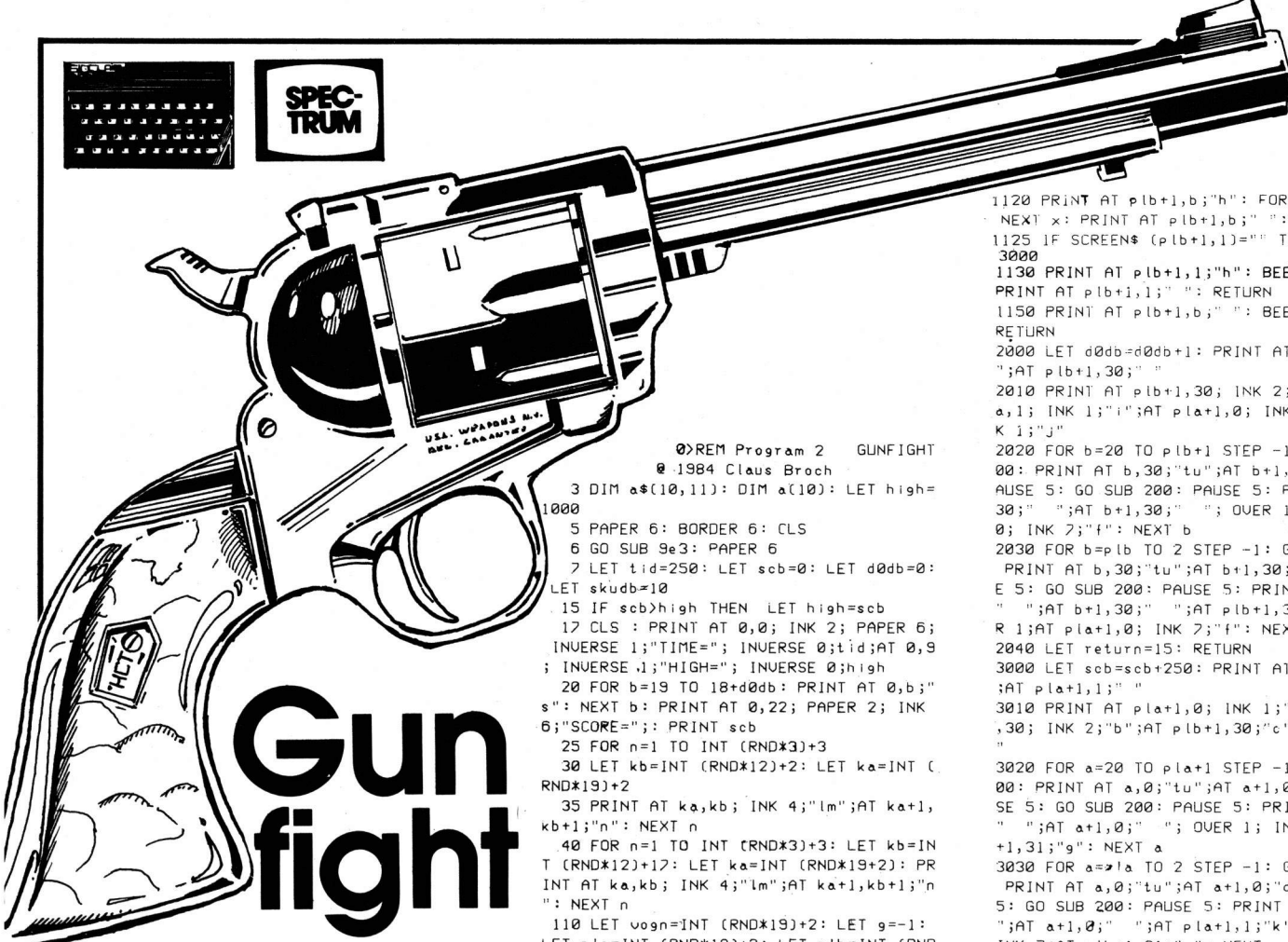
D (del) deler en linie. Dvs at al tekst efter cursoren bliver smidt ned på næste linie. Hvis den står midt i et ord, vil der blive sat en bindestreg. Derefter vil cursoren rykke en linie ned. Brugen af denne funktion er en effektiv måde at lave lige højremargen på.

Endelig er der ESC der stopper programmet. Den spørger dog lige først.

Programmet er, såvidt det nu har været muligt, sikret mod fejl. Men det kan dog godt ske at programmet stopper alligevel. Man kan så prøve at skrive CON nogle gange, men hvis man bliver ved med at få den samme fejl, er der ikke noget at gøre. Teksten er tabt. Hvis man får beskeden 'Diskette fejl' bør man indsætte en anden diskette. CON virker iverigt også, hvis man har stoppet programmet ved et uheld.

Tilbage er så kun at sige at tekstbehandling er et virkelig effektivt værktøj, specielt når man har sat sig ordentlig ind i det. Jeg ved ihvertfald med mig selv at jeg nu kan skrive stille ind på den halve tid, og med et dobbelt så pænt resultat!!□

Lars Christensen



Gunfight

Her får du din helt store udfordring i Gunfight, hvor du skal skyde en berygtet bandit fra Mexico. Programmet har 10 sværhedsgrader, og du får svært ved at skyde modstanderen i grad 10. Indtast først program Ø og run det. Derefter indtaster du program 1. Save programmet med "GOTO 9998". □

Claus Brock

```

1 REM Program 1
10 LET x=0: FOR n=USR "a" TO USR "a"+1
67: READ a: POKE n,a: LET x=x+a: NEXT n:
IF x<>11652 THEN PRINT FLASH 1;"FEJL I DATA": STOP
20 FOR n=144 TO 164: PRINT CHR$ n;: NEXT n: PRINT FLASH 1;"TRYK PAA EN TAST NAAR DU ER KLARTIL AT INDTASTE PROGRAM 2": PAUSE 0: NEW
30 DATA 0,16,56,56,56,56,124,0,0,112,248,112,112,32,112,120,116,119,116,88,76,68,68,102
40 DATA 8,47,47,47,63,32,32,32,16,244,244,244,252,4,4,4,0,3,4,8,8,8,8,8
50 DATA 0,192,32,16,16,16,16,16,0,252,0,0,0,0,0,0,14,31,14,4,14,30
60 DATA 46,238,46,26,50,34,34,102,24,24,126,126,24,24,24,24,0,0,8,28,28,31,31,15
70 DATA 32,112,112,112,112,240,240,242,119,119,127,127,126,112,112,112,0,0,0,0,0,0,135,255
80 DATA 32,112,32,32,32,4,222,254,4,14,4,4,4,32,123,127,0,0,0,0,0,225,255
90 DATA 0,60,90,126,60,36,60,0,0,1,0,0,63,127,127,127,128,192,128,128,254,255,255,255

```

```

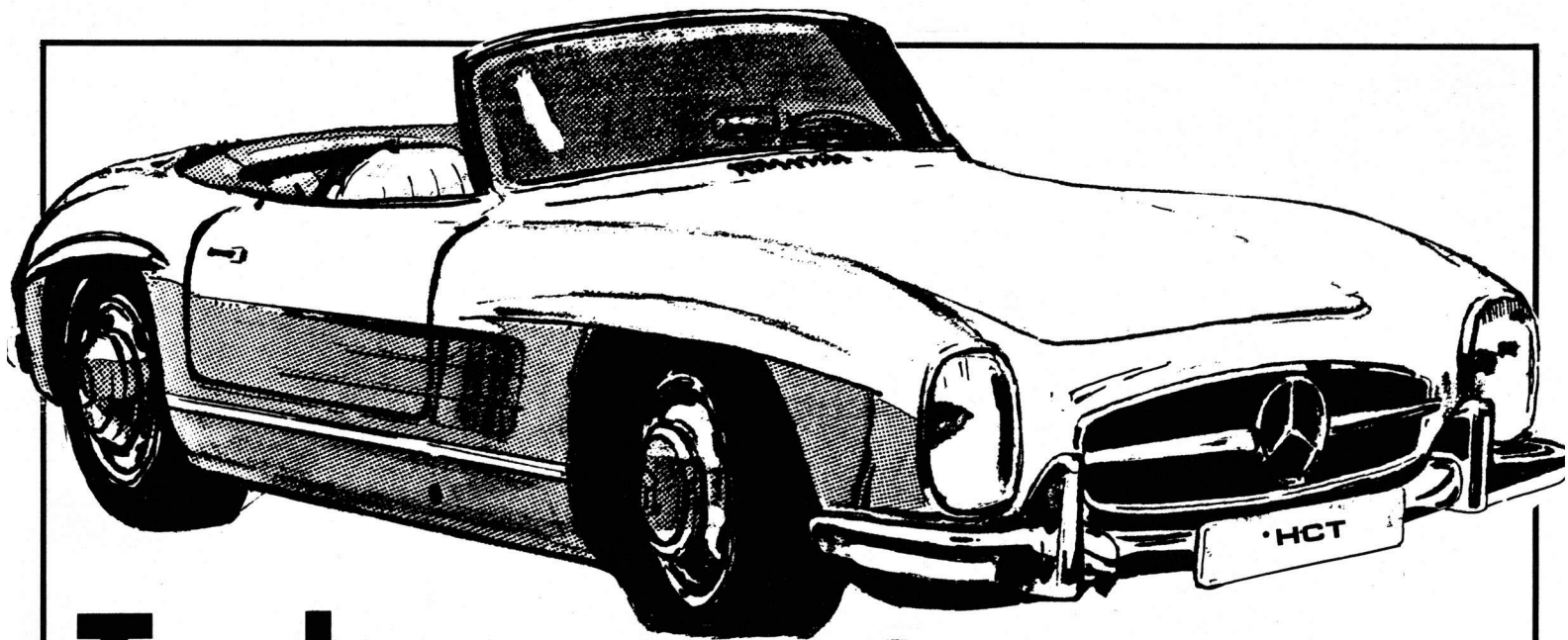
Ø)REM Program 2 GUNFIGHT
Ø 1984 Claus Brock
3 DIM a$(10,11): DIM a(10): LET high=1000
5 PAPER 6: BORDER 6: CLS
6 GO SUB 9e3: PAPER 6
7 LET tid=250: LET scb=0: LET d0db=0: LET skudb=10
15 IF scb>high THEN LET high=scb
17 CLS: PRINT AT 0,0: INK 2: PAPER 6: INVERSE 1;"TIME=": INVERSE 0;tid;AT 0,9: INVERSE 1;"HIGH=": INVERSE 0;high
20 FOR b=19 TO 18+d0db: PRINT AT 0,b;"s": NEXT b: PRINT AT 0,22: PAPER 2: INK 6;"SCORE=": PRINT scb
25 FOR n=1 TO INT (RND*3)+3
30 LET kb=INT (RND*12)+2: LET ka=INT (RND*19)+2
35 PRINT AT ka,kb: INK 4;"lm";AT ka+1,kb+1;"n": NEXT n
40 FOR n=1 TO INT (RND*3)+3: LET kb=INT (RND*12)+17: LET ka=INT (RND*19)+2: PRINT AT ka,kb: INK 4;"lm";AT ka+1,kb+1;"n": NEXT n
110 LET vogn=INT (RND*19)+2: LET g=-1: LET pla=INT (RND*19)+2: LET plb=INT (RND*19)+2: IF plb=pla THEN GO TO 110
111 GO SUB 200: PRINT AT pla,1: INK 1;"b";AT pla+1,1;"c";AT plb,30: INK 2;"i";AT plb+1,30;"j"
112 PRINT AT pla,1;" ";AT pla+1,1;" "
113 LET tid=tid-1: PRINT AT 0,5;tid;" "
: IF tid<=0 OR d0db>=3 THEN GO TO 4e3
115 LET pla=pla+(pla<plb AND RND(z)-(pl a>plb AND RND(z))
125 PRINT AT pla,1: INK 1;"b";AT pla+1,1;"c"
127 GO SUB 200
130 IF pla=plb+1 AND RND(z) OR pla=plb-2 AND RND(z) OR pla=plb AND RND(z) OR pla=plb-1 AND RND(z) THEN GO SUB 1e3: IF return=15 THEN GO TO return
134 PRINT AT plb,30;" ";AT plb+1,30;" "
135 LET plb=plb+(INKEY$="l" AND plb<20)-(INKEY$="0" AND plb>2)
139 PRINT AT plb,30: INK 2;"i";AT plb+1,30;"j"
140 IF INKEY$="q" THEN GO SUB 1100: IF return=15 THEN GO TO return
145 GO SUB 200: GO TO 112
200 PRINT AT vogn,15;" ";AT vogn+1,15;" "
205 IF vogn=20 THEN LET g=-1
210 IF vogn=2 THEN LET g=1
215 LET vogn=vogn+g
220 PRINT AT vogn,15: INK 7;"fg";AT vogn+1,15: INK 0;"de": RETURN
1000 LET return=0: IF RND>z THEN RETURN
1015 BEEP .01,0: BEEP .001,50: BEEP .003,0: BEEP .003,25: FOR a=2 TO 29: IF SCREEN$(pla+1,a)="" THEN GO TO 1050
1020 PRINT AT pla+1,a;"h": FOR x=1 TO 0: NEXT x: PRINT AT pla+1,a;" ": NEXT a
1025 IF SCREEN$(pla+1,30)="" THEN GO TO 0 2e3
1030 LET scb=scb+50: PRINT AT pla+1,30;"h";AT 0,28;scb: BEEP .01,50: PRINT AT pla+1,30;" ": RETURN
1050 PRINT AT pla+1,a;" ": BEEP .01,50: RETURN
1100 LET return=0
1110 BEEP .01,0: BEEP .001,50: BEEP .003,0: BEEP .003,25: FOR b=29 TO 2 STEP -1: IF SCREEN$(plb+1,b)="" THEN GO TO 1150

```

```

1120 PRINT AT plb+1,b;"h": FOR x=1 TO 0: NEXT x: PRINT AT plb+1,b;" ": NEXT b
1125 IF SCREEN$(plb+1,1)="" THEN GO TO 3000
1130 PRINT AT plb+1,1;"h": BEEP .01,50: PRINT AT plb+1,1;" ": RETURN
1150 PRINT AT plb+1,b;" ": BEEP .01,50: RETURN
2000 LET d0db=d0db+1: PRINT AT plb,30;" ";AT plb+1,30;" "
2010 PRINT AT plb+1,30: INK 2;"op";AT pl a,1: INK 1;"i";AT pla+1,0: INK 7;"f"; IN K 1;"j"
2020 FOR b=20 TO plb+1 STEP -1: GO SUB 2 00: PRINT AT b,30;"tu";AT b+1,30;"de": PAUSE 5: GO SUB 200: PAUSE 5: PRINT AT b,30;" ";AT b+1,30;" "; OVER 1;AT pla+1,0: INK 7;"f": NEXT b
2030 FOR b=plb TO 2 STEP -1: GO SUB 200: PRINT AT b,30;"tu";AT b+1,30;"de": PAUSE 5: GO SUB 200: PAUSE 5: PRINT AT b,30;" ";AT b+1,30;" ";AT plb+1,31;"k"; OVER 1;AT pla+1,1;" "
3010 PRINT AT pla+1,0: INK 1;"qr";AT plb,30: INK 2;"b";AT plb+1,30;"c"; INK 7;"g"
3020 FOR a=20 TO pla+1 STEP -1: GO SUB 2 00: PRINT AT a,0;"tu";AT a+1,0;"de": PAUSE 5: GO SUB 200: PAUSE 5: PRINT AT a,0;" ";AT a+1,0;" "; OVER 1: INK 7;AT plb +1,31;"g": NEXT a
3030 FOR a=pla TO 2 STEP -1: GO SUB 200: PRINT AT a,0;"tu";AT a+1,0;"de": PAUSE 5: GO SUB 200: PAUSE 5: PRINT AT a,0;" ";AT a+1,0;" ";AT pla+1,1;"k"; OVER 1: INK 7;AT plb+1,31;"g": NEXT a
3040 LET return=15: RETURN
4000 RESTORE: FOR n=1 TO 7: READ len,t o n,pau: BEEP len,ton: FOR p=0 TO pau: NEX T p: NEXT n: IF scb>high THEN LET high=scb
4005 PRINT AT 0,14;high;AT 11,11: PAPER 7: FLASH 1: BRIGHT 1;"GAME OVER"
4010 FOR n=0 TO 500: NEXT n: GO TO 6000
5000 DATA .1,25,0, len,15,0, len,5,0, len,5,0, len,5,0, len,15,0, len,25,0
6000 CLS
6010 FOR x=1 TO 10: IF scb<a(x) THEN NEXT x: GO TO 6500
6020 IF x<>10 THEN FOR i=10 TO x+1 STEP 1: LET a$(i)=a$(i-1): LET a(i)=a(i-1): NEXT i
6030 INPUT "Indtast dit navn ":"a$(x): LET a(x)=scb
6500 CLS: PRINT AT 0,14: INVERSE 1;"TOP -10"; INVERSE 0;AT 2,5;"Navn:";AT 2,20;"Score:";FOR k=1 TO 10: PRINT " "( TO 2-LEN (STR$ k));k;TAB 5;a$(k);TAB 20;"0 00"( TO 4-LEN (STR$ a(k)));a(k): NEXT k
6510 PRINT #1;"Tryk paa en tast for at spille": PAUSE 0: CLS: GO SUB 9490: GO TO 0 7
9000 PRINT AT 0,10: INK 2: PAPER 5;"INST RUKTION"
9001 PRINT : PRINT INK 1;" G UNFIGHT" " "I dette spil gaeld er det om at skyde sin modstander inden man bliver skudt af ham."
9002 PRINT INK 2;"Du styrer op med 'Ø' og ned med 'L'. Du skyder med 'Q'."
9003 PRINT INK 3;"Naar spillet starter er du til højre og skal saa skyde ham til venstre inden han skyder dig."
9004 PRINT INK 4;"Naar du enten er blevet skudt 3 gange eller tiden er løbet ud stopper spillet."
9100 FOR f=1 TO 10: LET a$(f)="Claus Brock": LET a(f)=1000: NEXT f
9490 PRINT "Indtast sværhedsgrad 1-10": INPUT z: IF z<1 OR z>10 THEN BEEP 1,3 5: GO TO 9490
9500 LET z=(z+(10-z)/2)/10: RETURN
9998 CLEAR: SAVE "gunfight" LINE 9999: SAVE "fightcode"CODE USR "a",168: VERIFY "": VERIFY ""CODE: STOP
9999 LOAD "fightcode"CODE USR "a": RUN

```



Turborace

Spillet er et racerbilspil, hvor du skal styre en lille bil rundt på en bane. Du må ikke støde ind i kanten, og hvis du kører ind i en isklot, skrider du ud.

I linie 10, 45, 510, 530, 580 og

600 er der USR-definerbare karakterer, hvor du så indtaster bogstavet i grafiskmode. Du bruger tasterne 6, 7, 8 og 9 til at flytte bilen.

Thøger Lyne



```

10 LET a$="facdhilleibig"
20 LET b$="323232212121"
30 LET c$="211332222123"
40 LET d$=" 8 8&99 7&97 6&76 6&8"
45 LET e$="acfhgdeb"
50 POKE 23624,7: PAPER 0: INK 7
60 CLS
70 LET o=0: GO SUB 1090
80 FOR a=1 TO 8
90 PRINT AT 3,8;"Brug Tasterne!"
100 PRINT AT a+5,11;a$(a);"--";d$(a*3);
d$(a*3+1);d$(a*3+2)
110 NEXT a
120 PRINT INK 4; FLASH 1;AT 21,16;"a'S
' For start d"
130 IF CODE INKEY$=115 THEN CLS : GO T
O 150
140 GO TO 130
150 LET t=40
160 LET s=0
170 LET x=5
180 LET y=15
190 LET c=1: GO SUB 200: GO SUB 840: GO
SUB 800: GO TO 200
200 RESTORE 1000
220 READ q
225 IF q=0 THEN RETURN
230 IF q>=100 THEN LET c=c+1: LET q=q-
100
240 PRINT INK 6;AT c,q;"@"
250 GO TO 220
260 PRINT AT x,y;a$(1)
270 LET c=4: GO SUB 280: GO TO 320
280 RESTORE 1200
290 FOR d=1 TO 97
300 LET a=INT (RND*26)+2
310 READ b
315 IF b>100 THEN LET c=c+1: LET b=b-1
00
320 IF b=a THEN PRINT INK 7;AT c,b;"*
"
330 NEXT d
340 LET f=INT ((256-IN 61438)/2)
350 IF f<1 OR f>12 THEN GO TO 430
355 IF f>7 AND x<9 THEN PRINT AT x,y;a
$(1): LET t=t-1/7: PRINT AT 1,1;INT t;"
": GO TO 340

```

```

360 PRINT AT x,y;" "
370 LET y=y-2+VAL b$(f)
380 LET x=x-2+VAL c$(f)
390 IF SCREEN$(x,y)="@" THEN GO TO 50
0
400 IF SCREEN$(x,y)="*" THEN GO SUB 6
40
410 PRINT INK 5;AT x,y;a$(f)
420 LET s=s+1/4
430 LET t=t-1/7
440 IF x<9 AND y=14 THEN LET s=s+10: L
ET y=15: GO SUB 800: PRINT AT x,y;a$(f)
450 IF INT t=0 THEN PRINT AT 1,11;INT
s;AT 1,1;INT t: GO TO 720
460 PRINT AT 1,11;INT s
470 PRINT AT 1,1;INT t;" "
480 GO TO 340
500 FOR a=1 TO 5
510 PRINT INK 6;AT x,y;"c"
520 BEEP .1,-25
530 PRINT INK 2;AT x,y;"b"
540 BEEP .2,-40
550 LET t=t-1
555 PRINT AT 1,1;INT t;" "
557 IF INT t=0 THEN GO TO 720
560 NEXT a
570 FOR a=1 TO 8
580 PRINT INK 3;AT x,y;"i"
590 BEEP .1,-50
600 PRINT INK 6;AT x,y;"i"
610 BEEP .2,-10
620 NEXT a
630 LET x=5: LET y=15: PRINT AT x,y;a$(
1): LET c=1: GO SUB 200: GO TO 340
640 FOR a=1 TO 8
650 PRINT AT x,y;e$(a)
660 BEEP .1,0
670 NEXT a
675 PRINT AT x,y;"*"
680 LET w=INT ((RND*2)-1)+x
690 LET r=INT ((RND*2)+1)+y
695 IF SCREEN$(w,r)="@" THEN GO TO 68
0
700 LET t=t-3
701 IF t<=0 THEN GO TO 720
705 LET x=w: LET y=r
710 RETURN

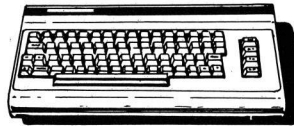
```

```

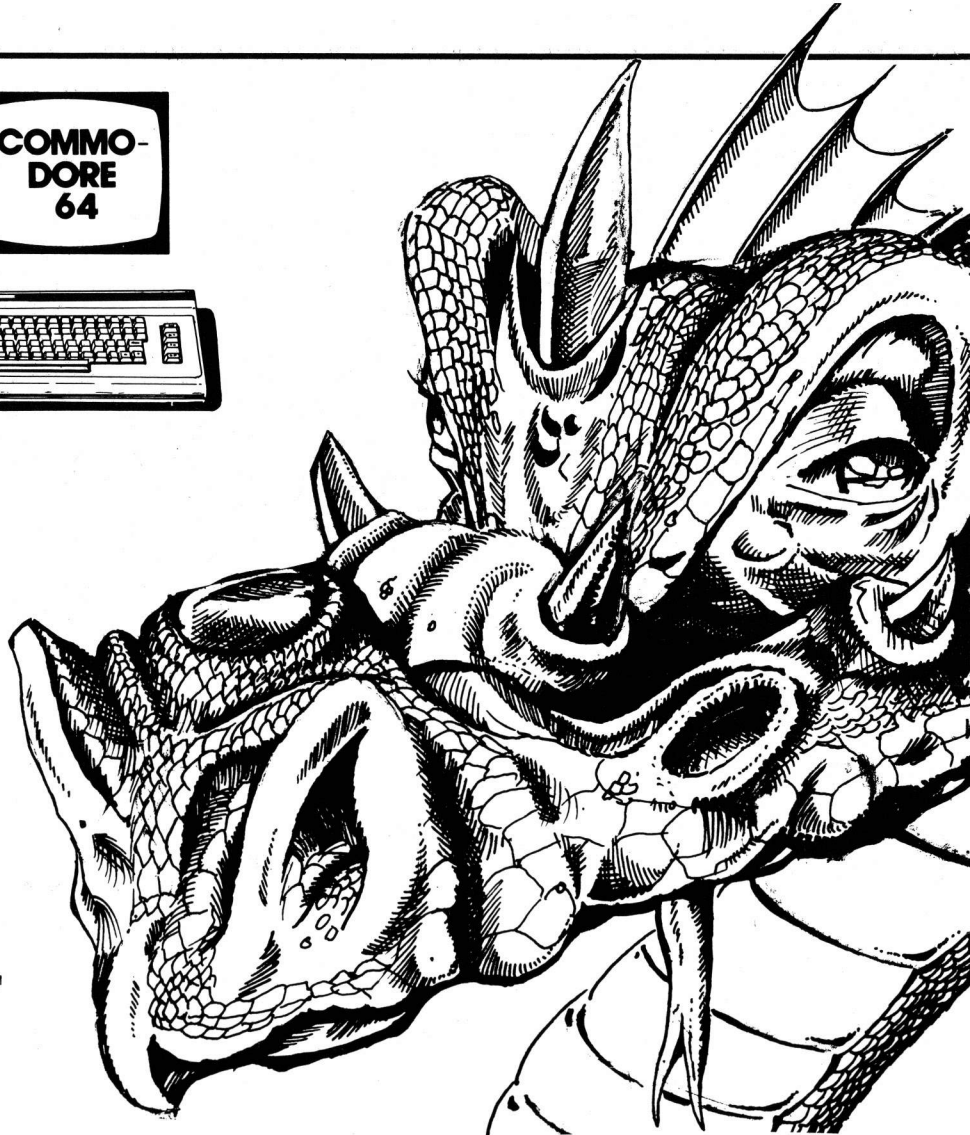
720 FOR a=1 TO 300: NEXT a: CLS
730 IF s>0 THEN PRINT AT 5,2;"Du slog
highscoren med ";INT s-INT o
735 IF s<0 THEN PRINT AT 5,4;"Du slog
ikke highscoren "
740 IF o<s THEN LET o=s
750 PRINT AT 8,5;"Highscoren er nu ";I
NT o
760 PRINT AT 12,3;"Uil du have et nyt s
pil ?"
770 IF CODE INKEY$=106 THEN CLS : GO T
O 80
780 IF CODE INKEY$=110 THEN STOP
790 GO TO 770
800 FOR h=4 TO 8
810 PRINT AT h,14;"!"
820 NEXT h
830 RETURN
840 PRINT INK 4; BRIGHT 1;AT 0,0;"Time
";AT 0,10;"Score";AT 0,18;"Highscore"; I
NK 7;AT 1,22;INT o
850 RETURN
1000 DATA 102,3,4,5,6,7,8,9,11,12,13,14,
15,16,17,18,19,20,21,22,23,24,25,26,27,1
01,2,5,7,9,10,11,14,16,18,23,27,101,27,2
8,101,28,29,101,2,29,30,102,6,7,8,9,30,1
02,6,7,9,10,11,12,21,22,29,30,101,2,6,12
,13,14,15,16,17,18,19,20,21,22,23,29,30;
101,6,23,24,30
1010 DATA 101,5,6,23,24,30,101,6,7,14,15
,16,17,18,23,30,101,7,8,11,12,13,14,18,1
9,20,23,24,30,101,8,9,10,11,20,21,22,23,
24,25,30,101,2,9,21,30,102,30,102,30,102
,29,30,102,3,14,15,16,17,18,21,22,23,24,
25,26,30,103,4,5,6,7,15,18,19,20,21,26,2
7,28,29,30,106,7,8,9,10,11,12,13,14,15,0
1090 RESTORE 1100
1100 DATA 24,60,60,24,24,90,126,90,208,2
24,116,186,29,46,20,8,11,7,46,93,184,180
,40,16,90,126,90,26,24,60,60,24
1110 DATA 0,7,98,255,255,98,7,0,0,224,70
,255,255,70,224,0,8,20,46,29,186,116,224
,208,16,40,116,184,93,46,7,11,32,224,104
,116,70,89,40,16
1115 FOR a=0 TO 72
1120 READ b
1130 POKE USR "a"+a,b
1140 NEXT a
1150 RETURN
1160 DATA 1
1200 DATA 3,5,7,9,10,11,12,17,19,20,22,2
5,103,4,6,8,10,12,16,18,21,23,26,104,8,1
2,17,20,23,25,26,27
1210 DATA 103,4,5,11,17,24,25,26,28,104,
5,25,26,27,104,26,3,4,26,27,103,26,27,28
,102,4,26,27,104,103,5,27,105,6,14,15,16
,27
1220 DATA 104,9,11,13,16,17,20,25,104,9,
11,13,14,15,16,17,19,20,25,105,8,12,15,1
8,21,23,27

```

Slimi



Slangen "Slimi" er en fætter til gode gamle Pacman. For hver runde bliver den frygtelige slange to led længere. Så gælder det om at se, hvor langt man kan nå, før man mister kontrollen. Spillet har 3 sværhedsgrader, og der kan enten benyttes joysticks eller Commodore 64's tastatur.



SEARCHING FOR \$\$

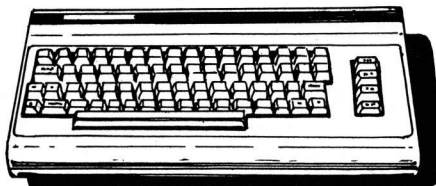
```

1 DIMF(2000):V=53281:HS=250:W$=" B,R"
2 DE=0
6 S1=0:S2=0:S3=0:S4=0:S5=0:S6=0:RE$="":LE$=""
7 GOSUB10000
8 GOSUB20000
9 GOSUB15000
15 S=0:P=2:E=0
20 K=0:O=170:PRINT"O":P=P+1
24 READL
25 L$=LEFT$(W$,4)
26 ON A0 GOSUB 3000,4000,5000
28 PRINT"#####"SPC(31);HS
29 PRINT"#####"SPC(35) ;(P-E)
30 K=0:O=170:X=0:GETC$:GETC$:ON OT GOT0400,500
400 J=PEEK(56320)
410 IF(JAND1)=0THENX=-40
420 IF(JAND2)=0THENX=40
430 IF(JAND4)=0THENX=-1
440 IF(JAND8)=0THENX=1
450 GOT0600
500 GETA$:IFA$="L"THENX=1
510 IFA$="K"THENX=-1
520 IFA$="O"THENX=-40
530 IFA$="R"THENX=40
600 F(O)=F(O-1)+X:IFPEEK(1106+F(O))=102THENK=2
610 IFPEEK(1106+F(O))=83THENS=S+1
620 IFPEEK(1106+F(O))=81THENK=2
700 POKE1106+F(O),81:POKE55378+F(O),1:POKE1106+F(O-L),96
710 PRINT"#####"SPC(31);S
750 IFK=2THEN900
760 O=O+1
770 IFAQ=3ANDS=(L/10)*134THENDE=DE+1:GOSUB30000:GOT020
780 IFAQ=2ANDS=(L/10)*123THENDE=DE+1:GOSUB30000:GOT020
790 IFAQ=1ANDS=(L/10)*89THENDE=DE+1:GOSUB30000:GOT020
800 ON OT GOT0 400,500
900 BA=54272:LA=15:FORHJ=60T040STEP-2:POKEBA+24,LA:POKEBA+6,15*16
910 POKEBA+1,HJ:POKEBA+4,17:FORCD=0T010:POKEV,CD:NEXTCD
911 POKE53280,0:POKE53281,0
912 LA=LA-1:NEXTHJ:POKEBA+24,0:POKEV,6
920 FORAA=0TOL:POKE1106+F(O-L)+AA,96:NEXTRA:POKEF(O)+1106,102
930 E=E+1:PRINT"#####"SPC(35) ;(P-E):IF(P-E)=0THEN2000
935 POKE55378+F(O),1:POKE53280,0:POKE53281,0:PRINT"$"
940 GOT0300
1000 DATA10,20,30,40,50,60,70,80,90,100,110,120,130,140,150,160,170
2000 IFS>HSTHENGOSUB50000
2010 RESTORE:GOSUB40000:GOT09
3000 POKE53280,0:POKE53281,0:PRINT"$"
3005 PRINT"#####"
3010 PRINT"#####" SCORE "
3020 PRINT"#####"
3030 PRINT"#####"
3040 PRINT"#####"
3050 PRINT"#####"
3060 PRINT"#####"
3070 PRINT"#####"
3080 PRINT"#####" HIGHSCORE
3090 PRINT"#####"
3100 PRINT"#####"
3110 PRINT"#####"
3120 PRINT"#####"
3130 PRINT"#####"
3140 PRINT"#####" BY
3150 PRINT"#####"
3160 PRINT"#####" :L$
3170 PRINT"#####"
3180 PRINT"#####" LIV - 3
3190 PRINT"#####"
3200 PRINT"#####"
3210 PRINT"#####"
3220 PRINT"#####"
3999 RETURN
  
```

```

4000 POKE53280,0:POKE53281,0:PRINT"$"
4005 PRINT"#####"
4010 PRINT"#####" SCORE "
4020 PRINT"#####"
4030 PRINT"#####"
4040 PRINT"#####"
4050 PRINT"#####"
4060 PRINT"#####"
4070 PRINT"#####" HIGHSCORE
4080 PRINT"#####"
4090 PRINT"#####"
4100 PRINT"#####"
4110 PRINT"#####"
4120 PRINT"#####"
4130 PRINT"#####"
4140 PRINT"#####" BY
4150 PRINT"#####"
4160 PRINT"#####" :L$
4170 PRINT"#####"
4180 PRINT"#####" LIV - 3
4190 PRINT"#####"
4200 PRINT"#####"
4210 PRINT"#####"
4220 PRINT"#####"
4999 RETURN
5000 POKE53280,0:POKE53281,0:PRINT"$"
5005 PRINT"#####"
5010 PRINT"#####" SCORE"
5020 PRINT"#####"
5030 PRINT"#####"
5040 PRINT"#####"
5050 PRINT"#####"
5060 PRINT"#####"
5070 PRINT"#####" HIGHSCORE
5080 PRINT"#####"
5090 PRINT"#####"
5100 PRINT"#####"
5110 PRINT"#####"
5120 PRINT"#####"
5130 PRINT"#####" BY
5140 PRINT"#####"
5150 PRINT"#####" :L$
5160 PRINT"#####"
5170 PRINT"#####" LIV - 3
5180 PRINT"#####"
5190 PRINT"#####"
5200 PRINT"#####"
5210 PRINT"#####"
5220 PRINT"#####"
5230 RETURN
  
```


Garfield II



I »Alt om Data« har vi tidligere vist en flot tegning af katten Garfield, fremstillet på en RC Piccolo computer.

I denne omgang vil vi vise dig, hvordan du med en VIC-20 kan få et vellignende portræt af den populære kat på din TV-skærm.

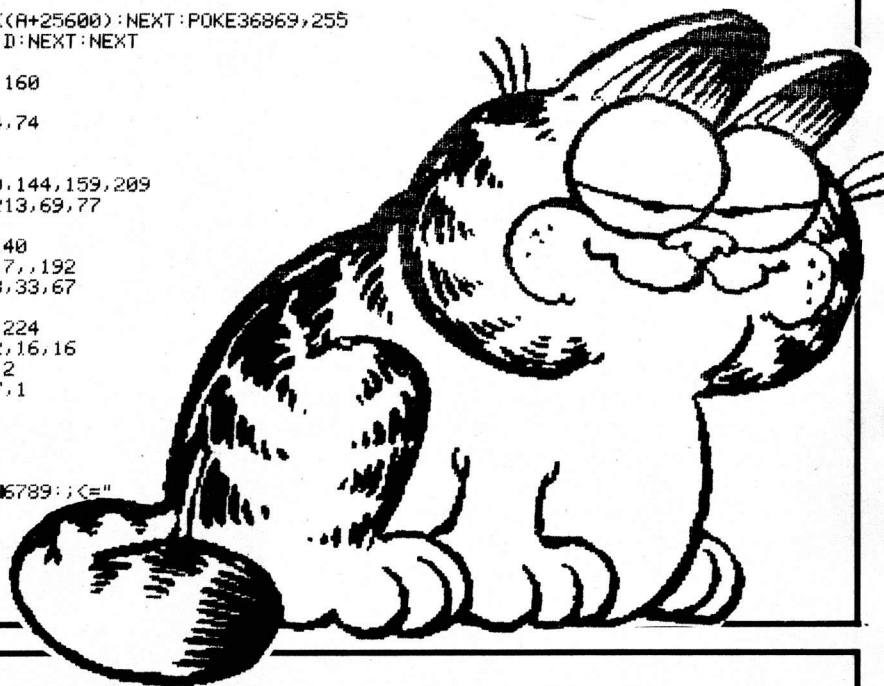
Programmet, der kører på en standard VIC'er, tegner Garfield i højopløsning. Og han ligner virkelig sig selv.

Hans Henrik
Hannibal Hansen

```

0 REM GARFIELD
1 REM HANS HENRIK H HANSEN
2 REM SILKEBORG 1984
10 PRINT " "
20 PRINT "  GARFIELD  "
30 PRINT "
40 PRINT "XXXXXXXXXX  COPYRIGHT 1984"
50 PRINT "  BY HANS HENRIK"
60 PRINT "XXXXX  ALL RIGHTS RESERVED"
70 PRINT "
80 FORT=1T05000:NEXT:PRINT"  HIT A KEY"
90 GETA$:IFA$=" "THEN90
100 PRINT"XXXXXXXXX  GARFIELD ER LAVET"
110 PRINT"  UD AF 46 DEFINERBARE"
120 PRINT"  KARAKTER"
130 PRINT"XXXXXX  POKE-VØRDIER FRA"
140 PRINT"  35-80"
150 FORT=1T04500:NEXT:PRINT"XXXXX  HIT A KEY"
160 GETA$:IFA$=" "THEN160
170 PRINT"XXXXXXXXX  HVIS BILLEDET SKAL"
180 PRINT"  LIGNE GARFIELD, SAA"
190 PRINT"  VØR SIKKER PAA"
200 PRINT"  AT DATA SØTNINGERNE"
210 PRINT"  ER OK, FØR RUN."
220 PRINT"  VENT LIDT CA (7 SEK.)"
230 FORT=1T03500:NEXT:PRINT"  HIT A KEY"
240 GETA$:IFA$=" "THEN240
250 PRINT" "
255 POKE 808,127
260 POKE52,28:POKE56,28:FORA=7448T07816:POKER,PEEK(A+25600):NEXT:POKE36869,255
270 FORA=0T045:B=FW8+7448:FORC=BT0B+7:READD:POKEC,D:NEXT:NEXT
280 DATA,,,,,1,,,,,129,,,7,25,35,68,137,8
290 DATA,,224,243,255,175,73,26,,,,,192,240,240,80,160
300 DATA,7,,6,9,,,,,70,44,156,124,64,199,200,210
310 DATA19,4,8,16,16,208,48,16,208,46,17,16,8,8,74,74
320 DATA64,128,195,196,105,82,84,73,,,,,192,,,192
330 DATA,,,,,60,67,64,64,64,1,1,1,57,69,133,133,132
340 DATA212,16,12,,192,224,240,224,16,16,8,199,160,144,159,209
350 DATA8,8,16,239,68,56,215,18,122,132,148,4,76,213,69,77
360 DATA32,,,,,112,136,9,14,8,,,,,224,16,8
370 DATA185,129,129,64,56,16,16,16,68,2,1,192,,,,,140
380 DATA192,64,64,254,30,14,3,,177,159,145,209,57,7,,192
390 DATA18,255,18,18,19,252,,,,,221,77,70,124,184,48,33,67
400 DATA12,4,4,24,,,,,3,128,8,16,240,8,8,48,192,32
410 DATA15,,,,,115,,,,,192,48,40,64,96,240,224
420 DATA,,120,230,1,,,,,134,152,96,64,32,32,16,16
430 DATA120,4,3,,,,,32,64,128,,,,,1,1,1,1,3,3,2
440 DATA192,,128,192,224,192,128,,231,79,12,,1,3,7,1
450 DATA,,128,128,128,128,128,128,8,8,4,4,4,4,4,4
460 DATA3,3,3,3,7,30,48,63,,128,192,128,,128,255
470 DATA,,4,2,1,1,34,255,240,8,68,34,18,20,36,255
480 DATA4,6,5,12,10,18,52,255,,,,,128,128,128,128,
490 A$="##%&XXXXXXXXXX)*,-XXXXXXXXXX/012345XXXXXXXXXXXX6789";C="
500 B$=">?~!~XXXXXXXXXXXXL I XXXXXXXXPLVTT"
510 PRINTSPC(118);A$
520 PRINTSPC(7);B$
530 GOTO530

```



```

*(11-I): NEXT I
52 LET R=11+INT (SUM/11)*11-SUM
53 IF R=10 THEN LET R=0
54 IF R=11 THEN LET R=1
55 PRINT : PRINT "DET SIDSTE CIFFER SK
AL VAERE "; FLASH 1;R: PRINT
56 IF NOT STR$ R=C$(10) THEN GO TO 60
57 IF (R/2)=INT (R/2) THEN PRINT "NUM
MERET PASSER PAA EN "; FLASH 1;"KVINDE"
58 IF (R/2)<>INT (R/2) THEN PRINT "NU
MMERET PASSER PAA EN "; FLASH 1;"MAND"
60 PRINT : PRINT s$
65 GO TO 15
75 LET MD=VAL C$(3 TO 4)
80 IF MD<1 OR MD>12 THEN GO TO 120

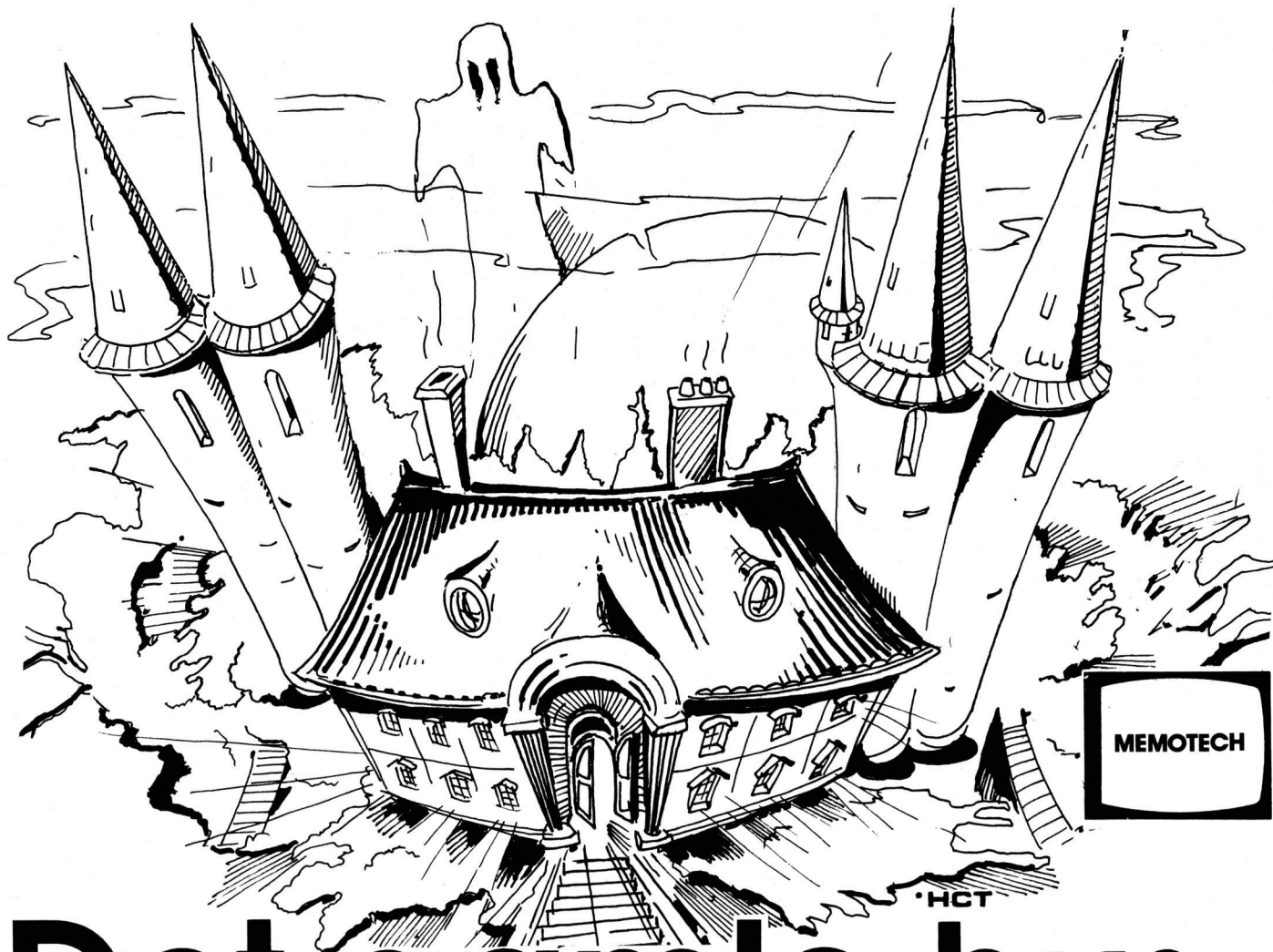
```

```

85 LET DA=VAL C$(1 TO 2)
90 IF (MD=1 OR MD=3 OR MD=5 OR MD=7 OR
MD=8 OR MD=10 OR MD=12) AND DA>31 THEN
GO TO 120
95 IF MD=2 AND DA>29 THEN GO TO 120
100 IF (MD=4 OR MD=6 OR MD=9 OR MD=11)
AND DA>30 THEN GO TO 120
105 LET SK=VAL C$(5 TO 6)/4
110 IF NOT SK=INT SK AND MD=2 AND DA>=2
9 THEN GO TO 120
115 GO TO 40
120 PRINT : PRINT FLASH 1;"DATOEN EKSI
STERER IKKE !"
125 PRINT s$: GO TO 15
9999 SAVE "CPR-TEST" LINE 0

```





Det gamle hus

"Det gamle hus" er et Adventure-(eventyr)spil, hvor du er fanget indeni en gammel faldetærdig rønne med en lige så for sømt have. Din opgave er at komme ud fra dette (farlige?) sted. Undervejs vil du støde på forskellige ting, som kan være til hjælp eller til besvær.

Nogle gode råd til løsning af spillet:

Din bedste hjælp er din fantasi.

Du kan skrive N. S. Ø. ell. V. når du vil gå mod henholdsvis

nord, syd, øst ell. vest.

Hvis du vil have hjælp, skriver du HJÆLP eller bare H, men prøv uden. En del af det sjove er jo at bruge fantasien til at finde ud af de ordrer, man kan bruge.

Du kan altid gemme spillet på et hvilket som helst tidspunkt.

Du skriver bare 'G' eller 'GEM'

Æ, Ø, Å sidder lige til venstre for linefeed-, BS- og ret-tasten <ret>.

LÆG

Bruges til at lægge en af de ting du bær på.

Skriv: LÆG <tal> eller L <tal>, hvor <tal> er nummeret på den ting, du vil lægge.

Eks.: LÆG 2 (lægger den anden ting på inventarlisten).

Retur: <R> + <ret>, næste ordre: <ret>

GEM

Med denne ordre kan du gemme "stillingen" på bånd, skriv 'G' eller 'GEM'.

TID

Med denne ordre kan du få at vide, hvor længe du har været i huset.

Retur: <R> + <ret>, næste ordre: <ret>

TAG

Denne ordre bruges til at ta' ting. Du skriver bare: TAG eller T, hvorved du tager den ting, der er i rummet.

INVENTAR

Med denne ordre kan du se hvad, du bær på, altså en inventarliste.

Skriv: INVENTAR eller I.

Retur: <R> + <ret>, næste ordre: <ret>

ÅBEN

Med denne ordre kan du åbne den ting der er i rummet.

Skriv: ÅBEN eller Å.

OPSPRÆT

Med denne ordre kan du sprætte noget op med.

Skriv: OPSPRÆT eller O.

Retur: <R> + <ret>, næste ordre: <ret>

MAD

Med denne ordre kan du made nogen eller noget.

Skriv: MAD eller M.

Peter Enø Cordsen

```

1 REM ** LINIE 10 ER AUTO-START **
10 SAVE "DET GAMLE HUS"
100 REM -----
110 REM * STYR *
120 REM -----
130 GOSUB 8000
140 PLOD "INTRO-STYR"
150 GOSUB 9000
160 GOSUB 200
185 CSR 1,22
190 INPUT "VIL DU PRØVE IGEN ? (J/N) ";F$
191 IF F$="J" THEN GOTO 150
199 STOP
200 REM -----
210 REM * INPUT-STYR *
220 REM -----
230 GOSUB 400
240 INPUT " HVAD VIL DU GØRE NU ? ";M$
250 PRINT : PRINT
260 IF M$="N" OR M$="S" OR M$="Ø" OR M$="V" THEN GOSUB 640
270 IF M$="T" OR LEFT$(M$,3)="TAG" THEN GOSUB 970
290 IF LEFT$(M$,1)="I" OR M$="INVENTAR" THEN GOSUB 1130
300 IF LEFT$(M$,1)="L" OR LEFT$(M$,3)="LÆG" THEN GOSUB 1280
310 IF M$="A" OR LEFT$(M$,4)="ABEN" OR M$="O" OR LEFT$(M$,7)="OPSPRÆT" THEN GOS
UB 1470
330 IF M$="M" OR LEFT$(M$,3)="MAD" THEN GOSUB 2000
340 IF M$="G" OR M$="GEM" THEN GOSUB 5000
350 IF M$="H" OR M$="HJÆLP" THEN PLOD "HJÆLP-STYR"
360 IF M$="TID" THEN GOSUB 4000
370 IF END$="SLUT" THEN RETURN
390 GOTO 200
400 REM -----
410 REM * START SPIL *
420 REM -----
440 CLS
450 PRINT " D E T G A M L E H U S ! "
460 PRINT " -----"
470 PRINT
475 PRINT
480 PRINT : PRINT
490 PRINT " DU ER I";R$(R)
500 PRINT
510 PRINT " DER ER UDGANGE MOD: ";
520 IF D1(R)<>0 THEN PRINT "NORD ";
530 IF D2(R)<>0 THEN PRINT "ØST ";
540 IF D3(R)<>0 THEN PRINT "SYD ";
550 IF D4(R)<>0 THEN PRINT "VEST ";
560 PRINT : PRINT
580 FOR I=1 TO 11
585 IF L(I)<>R THEN GOTO 600
590 PRINT " HER ER";N$(I);T$(I);
595 IF R=5 THEN PRINT " DEN ER "; ELSE PRINT " , DER ER ";
597 PRINT A$(I); PRINT
600 NEXT I
610 RETURN
640 REM -----
650 REM * N,Ø,S el. V *
660 REM -----
680 IF M$="N" AND D1(R)<>0 THEN LET R=D1(R); RETURN
700 IF M$="Ø" AND D2(R)<>0 THEN LET R=D2(R); RETURN
710 IF M$="S" AND D3(R)<>0 THEN LET R=D3(R); RETURN
720 IF M$="V" AND D4(R)<>0 THEN LET R=D4(R); RETURN
730 PRINT
740 PRINT " DU KAN IKKE GÅ MOD: ";
750 IF M$="N" THEN PRINT "NORD"
760 IF M$="Ø" THEN PRINT "ØST"
770 IF M$="S" THEN PRINT "SYD"
780 IF M$="V" THEN PRINT "VEST"
790 PAUSE 2500
800 RETURN
970 REM -----
980 REM * TAG ! *
990 REM -----
1020 FOR I=1 TO 11
1030 IF R=L(I) THEN GOTO 1055
1040 NEXT I
1050 PRINT " DER ER IKKE NOGET AT TA"; PAUSE 2500; RETURN
1055 IF R=1 AND B=0 THEN PRINT " BOKSEN ER LÆNKET TIL VÆGGEN !"; PAUSE 2500; R
ETURN
1057 IF R=3 AND RO=0 THEN PRINT " ROTTEN KLISTRER TIL GULVET !"; PAUSE 2500; R
ETURN
1060 IF R=10 AND P=0 THEN PRINT " PENGESKABET ER FOR TUNG"; PAUSE 2500; RETUR
N
1062 IF R=19 THEN PRINT " DU KAN DA IKKE TA' MONSTERET !"; PAUSE 2500; RETURN
1070 IF I=9 THEN LET F1=1
1080 IF I=2 THEN LET F2=1
1090 IF I=3 THEN LET F3=1
1095 IF I=1 THEN LET F4=1
1100 IF I=5 THEN LET F5=1
1110 IF I=4 THEN GOSUB 6000; RETURN
1112 LET L(I)=100
1115 PRINT " ";T$(I);" TAGET"
1120 PAUSE 2500; RETURN
1130 REM -----
1140 REM * INVENTAR ! *
1150 REM -----
1170 CLS
1180 PRINT " DU BÆR PÅ : "
1200 FOR I=1 TO 11
1210 IF L(I)<>100 THEN GOTO 1230
1215 PRINT
1220 PRINT " ";N$(I);T$(I)
1230 NEXT I
1240 PRINT : PRINT : PRINT : PRINT
1250 INPUT " TRYK <ret>";X$
1270 RETURN
1280 REM -----
1290 REM * LÆG ! *
1300 REM -----
1310 LET Y=0
1315 IF M$<>"L" AND M$<>"LÆG" THEN GOTO 1330
1320 PRINT " HVAD SKAL JEG LÆGGE ?"
1322 PAUSE 950
1324 PRINT : PRINT " >SE INVENTARLISTEN ELLER HJÆLP"
1327 PAUSE 2800; GOTO 1450
1330 IF LEFT$(M$,3)<>"LÆG" THEN LET I$=MID$(M$,3,3) ELSE LET I$=MID$(M$,5,5)
1340 FOR I=1 TO 11
1350 IF L(I)=100 THEN LET Y=Y+1
1360 IF L(I)=100 AND Y=VAL(I$) THEN GOTO 1410
1370 NEXT I
1380 PRINT : PRINT
1390 PRINT " DU BÆR IKKE PÅ NOGET SÅDANT"
1400 GOTO 1450
1410 IF I=9 THEN LET F1=0
1420 IF I=2 THEN LET F2=0
1430 IF I=3 THEN LET F3=0
1440 IF I=5 THEN LET F5=0
1443 LET L(I)=R
1445 PRINT " ";T$(I);" LAGT"
1450 PAUSE 2500; RETURN
1460 REM -----
1470 REM * ABEN, OPSPRÆT ! *
1480 REM -----
1490 IF R=1 AND B=0 THEN GOSUB 1600
1500 IF R=3 AND RO=0 THEN GOSUB 1700
1510 IF R=10 AND P=0 THEN GOSUB 1800
1520 IF R=20 THEN GOSUB 1900
1530 IF R<>1 AND B=0 AND R<>3 AND RO=0 AND R<>10 AND P=0 AND R<>20 THEN PRINT
" DER ER IKKE NOGET, JEG KAN ABNE HER "
1540 PAUSE 2800
1590 RETURN
1600 REM *****
1610 IF F2=0 THEN GOTO 1680
1620 PRINT " DER ER NOGLE NUMRE INDENI BOKSEN !"
1630 LET N$(1)="ET "
1640 LET T$(1)="STYKKE PAPIR"
1650 LET A$(1)="SKREVET NOGLE NUMRE PÅ"
1660 LET B=1
1670 RETURN
1680 IF R=1 THEN PRINT " JEG HAR IKKE NOGEN NØGLE"
1690 RETURN
1700 REM *****
1710 IF F1=0 THEN GOTO 1780
1720 PRINT " DER ER EN NØGLE INDENI DENS INDVOLDE!"
1730 LET T$(2)="NØGLE"
1740 LET A$(2)="AF GULD"
1750 LET RO=1
1760 RETURN
1780 PRINT " JEG HAR IKKE NOGEN KNIV"
1790 RETURN
1800 REM *****
1810 IF F4=0 THEN GOTO 1880
1820 PRINT " DER ER ET STYKKE STÅLTRÅD INDENI"
1822 PRINT : PRINT " PENGESKABET !"
1825 LET T$(5)="STYKKE STÅLTRÅD"
1830 LET A$(5)="STIPT"
1840 LET P=1
1850 RETURN
1880 PRINT " JEG KENDER IKKE KOMBINATIONEN"
1890 RETURN
1900 REM *****
1920 IF F1=1 AND F2=1 AND F3=1 AND F5=1 AND F6=1 THEN GOSUB 6500; RETURN
1930 PRINT " DU HAR IKKE GJORT NOK,"
1934 PRINT
1935 PAUSE 850
1936 PRINT " ELLER OGSÅ BÆR DU IKKE PÅ NOK"
1940 RETURN
2000 REM -----
2010 REM * MAD ! *
2020 REM -----
2030 IF R<>19 OR F6=1 THEN PRINT " DER ER INGEN, JEG KAN MADE"; GOTO 2100
2040 IF F3=0 THEN PRINT " JEG HAR INTET AT MADE MED"; GOTO 2100
2050 PRINT " MONSTERET KASTER OP OG DØR !"
2060 LET A$(10)="MEGET STORT OG DØDT !"
2070 LET F6=1
2100 PAUSE 2500
2110 RETURN
2400 REM -----
4010 REM * TID *
4020 REM -----
4040 PRINT " DU HAR NU VÆRET HERINDE I ";LEFT$(TIME$,2);";";MID$(TIME$,3,2);";"
;RIGHT$(TIME$,2)
4050 PAUSE 2700
4100 RETURN
5000 REM -----
5010 REM * SAVE ! *
5020 REM -----
5030 CLS
5040 INPUT " START BÅNDOPTAGEREN OG TRYK <ret>";F$
5043 CSR 3,6
5045 PRINT "NU GEMMES 'DET-GL-HUS' "
5050 SAVE "DET-GL-HUS"
5055 VS 0; CLS ; VS 5
5060 CSR 3,11
5062 PRINT "SPILLET ER NU GEMT"
5065 PAUSE 2100
5068 GOSUB 8000
5070 RETURN
6000 REM -----
6010 REM * DEADEND ! *
6020 REM -----
6025 LET END$="SLUT"
6030 CLS
6040 FOR I=1 TO 36
6050 PRINT "RAD LHCK I ";
6060 NEXT I
6065 PRINT
6070 PRINT : PRINT : PRINT : PRINT
6080 PRINT " PISTOLEN GIK AF I HÅNDEN PÅ DIG
OG DRÆBTE DIG !"
6085 PRINT
6090 FOR I=1 TO 5
6100 PRINT CHR$(7);
6110 NEXT I
6120 RETURN
6500 REM -----
6510 REM * UDEFOR ! *
6520 REM -----
6530 LET END$="SLUT"
6540 CLS
6550 FOR I=1 TO 100
6560 PRINT "TILLYKKE ! ";
6570 NEXT I
6580 PRINT : PRINT : PRINT
6590 PRINT " DU UNDSLAP PÅ ";LEFT$(TIME$,2);";";MID$(TIME$,3,2);";";RIG
HT$(TIME$,2);" !"
6591 REM PRINT " DU UNDSLAP PÅ ";LEFT$(TIME$,2);";TIMER ";MID$(TIME$,3,2)
;"; MINUTTER OG ";RIGHT$(TIME$,2);" SEKUNDER !"
6600 PRINT : PRINT : PRINT : PRINT
6610 RETURN
7030 RETURN
8000 REM -----
8010 REM * DANSKE BOGSTAVER *
8020 REM -----
8030 GENPAT 0,123,0,0,240,40,248,160,120,0
8040 GENPAT 0,91,56,80,144,144,248,144,154,0
8050 GENPAT 0,124,0,0,112,152,168,200,112,0
8060 GENPAT 0,92,112,136,152,168,200,136,112,0
8070 GENPAT 0,125,32,0,96,16,112,144,120,0
8080 GENPAT 0,93,32,0,32,80,136,248,136,0
8090 RETURN
9000 REM -----
9010 REM * INITIALISER *
9020 REM -----
9025 CLEAR : RESTORE 9180
9026 CLOCK "000000"
9027 LET Y=0; LET H$=""
9030 LET R=1; LET END$=""
9035 LET B=0; LET RO=0; LET P=0; LET M=0
9040 LET F1=0; LET F2=0; LET F3=0
9050 LET F4=0; LET F5=0; LET F6=0
9060 DIM R$(20,17),L(11)
9065 DIM D1(20),D2(20),D3(20),D4(20)

```

```

9070 DIM N$(11,4),T$(11,15),A$(11,35)
9080 FOR I=1 TO 20
9090 READ R$(I)
9100 NEXT I
9110 FOR I=1 TO 20
9120 READ D1(I),D2(I),D3(I),D4(I)
9130 NEXT I
9140 FOR I=1 TO 11
9150 READ N$(I),T$(I),A$(I),L(I)
9160 NEXT I
9165 REM
9170 REM ***** DATA FOR RUM *****
9175 REM
9180 DATA ET BADEVÆRELSE, EN SPISESTUE, ET KØKKEN, ET FORRÅDSKAMMER
9190 DATA ET SOVEVÆRELSE, EN KÆLDER, ET SKRALDERUM, ET COMPUTERRUM
9200 DATA ET MØRKT RUM, EN DAGLIGSTUE, ET LILLE RUM, ET STORT RUM
9210 DATA ET LEGERUM, EN SWIMMINGPOOL, ET DRIVHUS, EN HAVE
9220 DATA EN GARAGE, ET SKUR, ET STØVET RUM, EN ENTRE
9225 REM
9230 REM ***** DATA FOR DØRE *****
9235 REM
9240 DATA 0,2,0,0,0,3,0,1,0,4,0,2,0,0,7,3,0,6,0,0,0,0,9,5
9250 DATA 4,0,10,0,0,9,13,0,6,0,0,8,7,11,0,0,0,15,12,10
9260 DATA 11,0,19,14,8,14,0,0,0,12,17,13,0,0,0,11
9270 DATA 0,17,0,0,14,18,0,16,0,19,0,17,12,20,0,18,0,0,0,19
9275 REM
9280 REM ***** DATA FOR TING *****
9290 REM
9310 DATA EN .BOKS,LÅST,1, EN .ROTTE,DØD,3, EN .DÅSE,FULD AF CHAM- PIGNONS,4
9330 DATA EN .PISTOL,IKKE LADT,5
9340 DATA ET .PENGENSKAB,LÅST,10
9350 DATA EN .DØR,LÅST,20
9360 DATA EN .SKO,MEGET GAMMEL OG SNAVSET,14
9370 DATA EN .STIGE,BRÆKKET MIDT OVER,12
9380 DATA EN .KNIV,MEGET SKARP,16
9390 DATA ET .MONSTER,MEGET STORT,19
9400 DATA ET .SKELET,FRA EN SOM IKKE SLAP UD,17
9500 VS 0:CLS:VS 5:CLS
9900 RETURN
9920 REM *****
9922 CLS:INPUT " SÆT TOP":X$
9924 LPRINT CHR$(27);"("
9926 LPRINT CHR$(27);"1":CHR$(41)
9930 PLOD "UD1"
9932 PLOD "UD2"
9934 LPRINT CHR$(27);"1":CHR$(0)
9936 LPRINT CHR$(27);"N":CHR$(1)
9937 LPRINT CHR$(12)
9938 LLIST
9940 STOP

```

Oversigt

H HJÆLP
G GEM
TID
T TAG
I INVENTAR

L LÆG (nr. på Inventar-listen)
Å ÅBEN
O OPSPRÆT
M MAD
N S Ø V for nord syd øst vest
NB. afslut altid med <ret>

Hjælp-styr

~b *DISPLAY VALG *E
*IF R, r
*IF O, o
*GOTO H1.
~h *D INSTRUK2. *E *B b
~o *D OVERSIGT *E *B b
~r *RETURN

H1

*IF GEM, g
*IF TID, g
*IF TAG, t
*IF INVENTARLISTE, t
*IF INVENTAR, t
*IF LÆG, l
*IF ÅBEN, å
*IF OPSPRÆT, å
*IF MAD, m
~l *DISPLAY INSTRUK1. *E *I R, b
~g *DISPLAY INSTRUK2. *E *I R, b
~t *DISPLAY INSTRUK3. *E *I R, b
~å *DISPLAY INSTRUK4. *E *I R, b
~m *DISPLAY INSTRUK5. *E
~b *GOTO HJÆLP-STYR.

Hjælp!

Hvad vil du se

- En ordreoversigt: <O> + <ret>
- Den første ordre: <ret>
- En bestemt ordre: <R> + <ret>
Vil du retur til spil: <R> + <ret>

Kære læser.

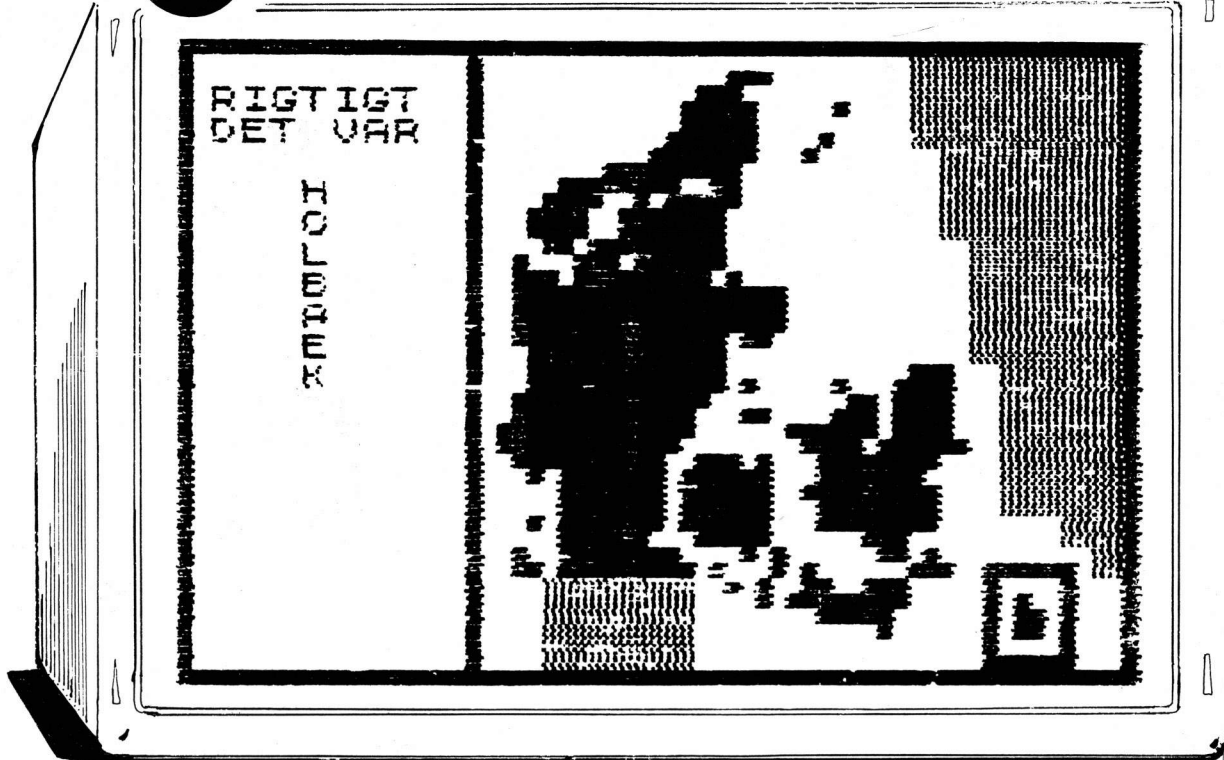
Vi håber, at du er tilfreds med programmerne i det første nummer af SOFT SPECIAL.

Skulle du ligge inde med tilsvarende gode – og måske endnu sjovere og bedre – programmer, du selv har lavet, hører vi gerne fra dig. SOFT er nemlig læsernes blad, og vi betaler op til 1000 kroner for et rigtig spændende program. Og måske får netop dit program en fremtrædende plads i det næste SOFT SPECIAL eller i "Alt om Data".

Indsend dit program på kassettebånd eller disc og gerne vedlagt en udprintning. Samtlige programmer testes og vurderes, inden de offentliggøres. Alt materiale returneres efter brug, og bladet har fuld ret til at publicere programmet på tryk og eventuelt andet lagermedium.

Programmer sendes til "Alt om Data", St. Kongensgade 72, 1264 København K.

Geografi



```

10 DIM O$(1,15)
11 GOTO 170
12 REM "INDTASTNING AF BYER"
13 DIM A$(50)
14 DIM B$(50)
15 DIM C$(50,15)
16 DIM B#$(50,33)
17 FOR N=1 TO 50
18 PRINT "BY "N;" = "
19 INPUT A$(N)
20 PRINT "1. KOORDINATOR ="
21 INPUT A(N)
22 PRINT "2. KOORDINATOR ="
23 INPUT B(N)
24 PRINT "DATA OM BYEN ="
25 INPUT B#(N)
26 CLS
27 NEXT N
28 PRINT "INDTASTNING FÆRDIG"
29 STOP
30 LET P=0
31 LET C=0
32 REM INSTRUKTION
33 PRINT AT 0,10;"GEOGRAFI"
34 "DETTE PROGRAM OMHANDLER EMNET"
35 "GEOGRAFI."
36 "MÅSKEN UDVEJL"
37 "ER EN TILFÆLDIGSTORRE BY VED AT"
38 "SÆTTE EN PRIK."
39 "DET ER NU DI"
40 "N OPGAVE AT FINDE"
41 "NAVNET PÅ B"
42 "YEN."
43 PRINT AT 12,0;"TAST N/L FOR"
44 "AT SE SPECIELLE STAVEMÅDER"
45 IF INKEY$="" THEN GOTO 220
46 CLS
47 PRINT AT 0,10;"GEOGRAFI"
48 "SPECIELLE STAVEMÅDER:"
49 "KØBING SJ"
50 "NYKØBING F"
51 "ÆBELT"
52 "FT"
53 "NÆSTVED"
54 "ÅLBORG"
55 "OS"
56 "AT 13,0;"TAST N/L FOR AT SE"
57 "YENDE"
58 IF INKEY$="" THEN GOTO 250
59 CLS
60 REM "TEGNING AF DANMARKSKOR"
61 "TET"
62 PRINT ..
63 PRINT ..
64 PRINT ..
65 PRINT ..
66 PRINT ..
67 PRINT ..
68 PRINT ..
69 PRINT ..
70 PRINT ..
71 PRINT ..
72 PRINT ..
73 PRINT ..
74 PRINT ..
75 PRINT ..
76 PRINT ..
77 PRINT ..
78 PRINT ..
79 PRINT ..
80 PRINT ..
81 PRINT ..
82 PRINT ..
83 PRINT ..
84 PRINT ..
85 PRINT ..
86 PRINT ..
87 PRINT ..
88 PRINT ..
89 PRINT ..
90 PRINT ..
91 PRINT ..
92 PRINT ..
93 PRINT ..
94 PRINT ..
95 PRINT ..
96 PRINT ..
97 PRINT ..
98 PRINT ..
99 PRINT ..
100 PRINT ..
101 PRINT ..
102 PRINT ..
103 PRINT ..
104 PRINT ..
105 PRINT ..
106 PRINT ..
107 PRINT ..
108 PRINT ..
109 PRINT ..
110 PRINT ..
111 PRINT ..
112 PRINT ..
113 PRINT ..
114 PRINT ..
115 PRINT ..
116 PRINT ..
117 PRINT ..
118 PRINT ..
119 PRINT ..
120 PRINT ..
121 PRINT ..
122 PRINT ..
123 PRINT ..
124 PRINT ..
125 PRINT ..
126 PRINT ..
127 PRINT ..
128 PRINT ..
129 PRINT ..
130 PRINT ..
131 PRINT ..
132 PRINT ..
133 PRINT ..
134 PRINT ..
135 PRINT ..
136 PRINT ..
137 PRINT ..
138 PRINT ..
139 PRINT ..
140 PRINT ..
141 PRINT ..
142 PRINT ..
143 PRINT ..
144 PRINT ..
145 PRINT ..
146 PRINT ..
147 PRINT ..
148 PRINT ..
149 PRINT ..
150 PRINT ..
151 PRINT ..
152 PRINT ..
153 PRINT ..
154 PRINT ..
155 PRINT ..
156 PRINT ..
157 PRINT ..
158 PRINT ..
159 PRINT ..
160 PRINT ..
161 PRINT ..
162 PRINT ..
163 PRINT ..
164 PRINT ..
165 PRINT ..
166 PRINT ..
167 PRINT ..
168 PRINT ..
169 PRINT ..
170 PRINT ..
171 PRINT ..
172 PRINT ..
173 PRINT ..
174 PRINT ..
175 PRINT ..
176 PRINT ..
177 PRINT ..
178 PRINT ..
179 PRINT ..
180 PRINT ..
181 PRINT ..
182 PRINT ..
183 PRINT ..
184 PRINT ..
185 PRINT ..
186 PRINT ..
187 PRINT ..
188 PRINT ..
189 PRINT ..
190 PRINT ..
191 PRINT ..
192 PRINT ..
193 PRINT ..
194 PRINT ..
195 PRINT ..
196 PRINT ..
197 PRINT ..
198 PRINT ..
199 PRINT ..
200 PRINT ..
201 PRINT ..
202 PRINT ..
203 PRINT ..
204 PRINT ..
205 PRINT ..
206 PRINT ..
207 PRINT ..
208 PRINT ..
209 PRINT ..
210 PRINT ..
211 PRINT ..
212 PRINT ..
213 PRINT ..
214 PRINT ..
215 PRINT ..
216 PRINT ..
217 PRINT ..
218 PRINT ..
219 PRINT ..
220 PRINT ..
221 PRINT ..
222 PRINT ..
223 PRINT ..
224 PRINT ..
225 PRINT ..
226 PRINT ..
227 PRINT ..
228 PRINT ..
229 PRINT ..
230 PRINT ..
231 PRINT ..
232 PRINT ..
233 PRINT ..
234 PRINT ..
235 PRINT ..
236 PRINT ..
237 PRINT ..
238 PRINT ..
239 PRINT ..
240 PRINT ..
241 PRINT ..
242 PRINT ..
243 PRINT ..
244 PRINT ..
245 PRINT ..
246 PRINT ..
247 PRINT ..
248 PRINT ..
249 PRINT ..
250 PRINT ..
251 PRINT ..
252 PRINT ..
253 PRINT ..
254 PRINT ..
255 PRINT ..
256 PRINT ..
257 PRINT ..
258 PRINT ..
259 PRINT ..
260 PRINT ..
261 PRINT ..
262 PRINT ..
263 PRINT ..
264 PRINT ..
265 PRINT ..
266 PRINT ..
267 PRINT ..
268 PRINT ..
269 PRINT ..
270 PRINT ..
271 PRINT ..
272 PRINT ..
273 PRINT ..
274 PRINT ..
275 PRINT ..
276 PRINT ..
277 PRINT ..
278 PRINT ..
279 PRINT ..
280 PRINT ..
281 PRINT ..
282 PRINT ..
283 PRINT ..
284 PRINT ..
285 PRINT ..
286 PRINT ..
287 PRINT ..
288 PRINT ..
289 PRINT ..
290 PRINT ..
291 PRINT ..
292 PRINT ..
293 PRINT ..
294 PRINT ..
295 PRINT ..
296 PRINT ..
297 PRINT ..
298 PRINT ..
299 PRINT ..
300 PRINT ..
301 PRINT ..
302 PRINT ..
303 PRINT ..
304 PRINT ..
305 PRINT ..
306 PRINT ..
307 PRINT ..
308 PRINT ..
309 PRINT ..
310 PRINT ..
311 PRINT ..
312 PRINT ..
313 PRINT ..
314 PRINT ..
315 PRINT ..
316 PRINT ..
317 PRINT ..
318 PRINT ..
319 PRINT ..
320 PRINT ..
321 PRINT ..
322 PRINT ..
323 PRINT ..
324 PRINT ..
325 PRINT ..
326 PRINT ..
327 PRINT ..
328 PRINT ..
329 PRINT ..
330 PRINT ..
331 PRINT ..
332 PRINT ..
333 PRINT ..
334 PRINT ..
335 PRINT ..
336 PRINT ..
337 PRINT ..
338 PRINT ..
339 PRINT ..
340 PRINT ..
341 PRINT ..
342 PRINT ..
343 PRINT ..
344 PRINT ..
345 PRINT ..
346 PRINT ..
347 PRINT ..
348 PRINT ..
349 PRINT ..
350 PRINT ..
351 PRINT ..
352 PRINT ..
353 PRINT ..
354 PRINT ..
355 PRINT ..
356 PRINT ..
357 PRINT ..
358 PRINT ..
359 PRINT ..
360 PRINT ..
361 PRINT ..
362 PRINT ..
363 PRINT ..
364 PRINT ..
365 PRINT ..
366 PRINT ..
367 PRINT ..
368 PRINT ..
369 PRINT ..
370 PRINT ..
371 PRINT ..
372 PRINT ..
373 PRINT ..
374 PRINT ..
375 PRINT ..
376 PRINT ..
377 PRINT ..
378 PRINT ..
379 PRINT ..
380 PRINT ..
381 PRINT ..
382 PRINT ..
383 PRINT ..
384 PRINT ..
385 PRINT ..
386 PRINT ..
387 PRINT ..
388 PRINT ..
389 PRINT ..
390 PRINT ..
391 PRINT ..
392 PRINT ..
393 PRINT ..
394 PRINT ..
395 PRINT ..
396 PRINT ..
397 PRINT ..
398 PRINT ..
399 PRINT ..
400 PRINT ..
401 PRINT ..
402 PRINT ..
403 PRINT ..
404 PRINT ..
405 PRINT ..
406 PRINT ..
407 PRINT ..
408 PRINT ..
409 PRINT ..
410 PRINT ..
411 PRINT ..
412 PRINT ..
413 PRINT ..
414 PRINT ..
415 PRINT ..
416 PRINT ..
417 PRINT ..
418 PRINT ..
419 PRINT ..
420 PRINT ..
421 PRINT ..
422 PRINT ..
423 PRINT ..
424 PRINT ..
425 PRINT ..
426 PRINT ..
427 PRINT ..
428 PRINT ..
429 PRINT ..
430 PRINT ..
431 PRINT ..
432 PRINT ..
433 PRINT ..
434 PRINT ..
435 PRINT ..
436 PRINT ..
437 PRINT ..
438 PRINT ..
439 PRINT ..
440 PRINT ..
441 PRINT ..
442 PRINT ..
443 PRINT ..
444 PRINT ..
445 PRINT ..
446 PRINT ..
447 PRINT ..
448 PRINT ..
449 PRINT ..
450 PRINT ..
451 PRINT ..
452 PRINT ..
453 PRINT ..
454 PRINT ..
455 PRINT ..
456 PRINT ..
457 PRINT ..
458 PRINT ..
459 PRINT ..
460 PRINT ..
461 PRINT ..
462 PRINT ..
463 PRINT ..
464 PRINT ..
465 PRINT ..
466 PRINT ..
467 PRINT ..
468 PRINT ..
469 PRINT ..
470 PRINT ..
471 PRINT ..
472 PRINT ..
473 PRINT ..
474 PRINT ..
475 PRINT ..
476 PRINT ..
477 PRINT ..
478 PRINT ..
479 PRINT ..
480 PRINT ..
481 PRINT ..
482 PRINT ..
483 PRINT ..
484 PRINT ..
485 PRINT ..
486 PRINT ..
487 PRINT ..
488 PRINT ..
489 PRINT ..
490 PRINT ..
491 PRINT ..
492 PRINT ..
493 PRINT ..
494 PRINT ..
495 PRINT ..
496 PRINT ..
497 PRINT ..
498 PRINT ..
499 PRINT ..
500 PRINT ..
501 PRINT ..
502 PRINT ..
503 PRINT ..
504 PRINT ..
505 PRINT ..
506 PRINT ..
507 PRINT ..
508 PRINT ..
509 PRINT ..
510 PRINT ..
511 PRINT ..
512 PRINT ..
513 PRINT ..
514 PRINT ..
515 PRINT ..
516 PRINT ..
517 PRINT ..
518 PRINT ..
519 PRINT ..
520 PRINT ..
521 PRINT ..
522 PRINT ..
523 PRINT ..
524 PRINT ..
525 PRINT ..
526 PRINT ..
527 PRINT ..
528 PRINT ..
529 PRINT ..
530 PRINT ..
531 PRINT ..
532 PRINT ..
533 PRINT ..
534 PRINT ..
535 PRINT ..
536 PRINT ..
537 PRINT ..
538 PRINT ..
539 PRINT ..
540 PRINT ..
541 PRINT ..
542 PRINT ..
543 PRINT ..
544 PRINT ..
545 PRINT ..
546 PRINT ..
547 PRINT ..
548 PRINT ..
549 PRINT ..
550 PRINT ..
551 PRINT ..
552 PRINT ..
553 PRINT ..
554 PRINT ..
555 PRINT ..
556 PRINT ..
557 PRINT ..
558 PRINT ..
559 PRINT ..
560 PRINT ..
561 PRINT ..
562 PRINT ..
563 PRINT ..
564 PRINT ..
565 PRINT ..
566 PRINT ..
567 PRINT ..
568 PRINT ..
569 PRINT ..
570 PRINT ..
571 PRINT ..
572 PRINT ..
573 PRINT ..
574 PRINT ..
575 PRINT ..
576 PRINT ..
577 PRINT ..
578 PRINT ..
579 PRINT ..
580 PRINT ..
581 PRINT ..
582 PRINT ..
583 PRINT ..
584 PRINT ..
585 PRINT ..
586 PRINT ..
587 PRINT ..
588 PRINT ..
589 PRINT ..
590 PRINT ..
591 PRINT ..
592 PRINT ..
593 PRINT ..
594 PRINT ..
595 PRINT ..
596 PRINT ..
597 PRINT ..
598 PRINT ..
599 PRINT ..
600 PRINT ..
601 PRINT ..
602 PRINT ..
603 PRINT ..
604 PRINT ..
605 PRINT ..
606 PRINT ..
607 PRINT ..
608 PRINT ..
609 PRINT ..
610 PRINT ..
611 PRINT ..
612 PRINT ..
613 PRINT ..
614 PRINT ..
615 PRINT ..
616 PRINT ..
617 PRINT ..
618 PRINT ..
619 PRINT ..
620 PRINT ..
621 PRINT ..
622 PRINT ..
623 PRINT ..
624 PRINT ..
625 PRINT ..
626 PRINT ..
627 PRINT ..
628 PRINT ..
629 PRINT ..
630 PRINT ..
631 PRINT ..
632 PRINT ..
633 PRINT ..
634 PRINT ..
635 PRINT ..
636 PRINT ..
637 PRINT ..
638 PRINT ..
639 PRINT ..
640 PRINT ..
641 PRINT ..
642 PRINT ..
643 PRINT ..
644 PRINT ..
645 PRINT ..
646 PRINT ..
647 PRINT ..
648 PRINT ..
649 PRINT ..
650 PRINT ..
651 PRINT ..
652 PRINT ..
653 PRINT ..
654 PRINT ..
655 PRINT ..
656 PRINT ..
657 PRINT ..
658 PRINT ..
659 PRINT ..
660 PRINT ..
661 PRINT ..
662 PRINT ..
663 PRINT ..
664 PRINT ..
665 PRINT ..
666 PRINT ..
667 PRINT ..
668 PRINT ..
669 PRINT ..
670 PRINT ..
671 PRINT ..
672 PRINT ..
673 PRINT ..
674 PRINT ..
675 PRINT ..
676 PRINT ..
677 PRINT ..
678 PRINT ..
679 PRINT ..
680 PRINT ..
681 PRINT ..
682 PRINT ..
683 PRINT ..
684 PRINT ..
685 PRINT ..
686 PRINT ..
687 PRINT ..
688 PRINT ..
689 PRINT ..
690 PRINT ..
691 PRINT ..
692 PRINT ..
693 PRINT ..
694 PRINT ..
695 PRINT ..
696 PRINT ..
697 PRINT ..
698 PRINT ..
699 PRINT ..
700 PRINT ..
701 PRINT ..
702 PRINT ..
703 PRINT ..
704 PRINT ..
705 PRINT ..
706 PRINT ..
707 PRINT ..
708 PRINT ..
709 PRINT ..
710 PRINT ..
711 PRINT ..
712 PRINT ..
713 PRINT ..
714 PRINT ..
715 PRINT ..
716 PRINT ..
717 PRINT ..
718 PRINT ..
719 PRINT ..
720 PRINT ..
721 PRINT ..
722 PRINT ..
723 PRINT ..
724 PRINT ..
725 PRINT ..
726 PRINT ..
727 PRINT ..
728 PRINT ..
729 PRINT ..
730 PRINT ..
731 PRINT ..
732 PRINT ..
733 PRINT ..
734 PRINT ..
735 PRINT ..
736 PRINT ..
737 PRINT ..
738 PRINT ..
739 PRINT ..
740 PRINT ..
741 PRINT ..
742 PRINT ..
743 PRINT ..
744 PRINT ..
745 PRINT ..
746 PRINT ..
747 PRINT ..
748 PRINT ..
749 PRINT ..
750 PRINT ..
751 PRINT ..
752 PRINT ..
753 PRINT ..
754 PRINT ..
755 PRINT ..
756 PRINT ..
757 PRINT ..
758 PRINT ..
759 PRINT ..
760 PRINT ..
761 PRINT ..
762 PRINT ..
763 PRINT ..
764 PRINT ..
765 PRINT ..
766 PRINT ..
767 PRINT ..
768 PRINT ..
769 PRINT ..
770 PRINT ..
771 PRINT ..
772 PRINT ..
773 PRINT ..
774 PRINT ..
775 PRINT ..
776 PRINT ..
777 PRINT ..
778 PRINT ..
779 PRINT ..
780 PRINT ..
781 PRINT ..
782 PRINT ..
783 PRINT ..
784 PRINT ..
785 PRINT ..
786 PRINT ..
787 PRINT ..
788 PRINT ..
789 PRINT ..
790 PRINT ..
791 PRINT ..
792 PRINT ..
793 PRINT ..
794 PRINT ..
795 PRINT ..
796 PRINT ..
797 PRINT ..
798 PRINT ..
799 PRINT ..
800 PRINT ..
801 PRINT ..
802 PRINT ..
803 PRINT ..
804 PRINT ..
805 PRINT ..
806 PRINT ..
807 PRINT ..
808 PRINT ..
809 PRINT ..
810 PRINT ..
811 PRINT ..
812 PRINT ..
813 PRINT ..
814 PRINT ..
815 PRINT ..
816 PRINT ..
817 PRINT ..
818 PRINT ..
819 PRINT ..
820 PRINT ..
821 PRINT ..
822 PRINT ..
823 PRINT ..
824 PRINT ..
825 PRINT ..
826 PRINT ..
827 PRINT ..
828 PRINT ..
829 PRINT ..
830 PRINT ..
831 PRINT ..
832 PRINT ..
833 PRINT ..
834 PRINT ..
835 PRINT ..
836 PRINT ..
837 PRINT ..
838 PRINT ..
839 PRINT ..
840 PRINT ..
841 PRINT ..
842 PRINT ..
843 PRINT ..
844 PRINT ..
845 PRINT ..
846 PRINT ..
847 PRINT ..
848 PRINT ..
849 PRINT ..
850 PRINT ..
851 PRINT ..
852 PRINT ..
853 PRINT ..
854 PRINT ..
855 PRINT ..
856 PRINT ..
857 PRINT ..
858 PRINT ..
859 PRINT ..
860 PRINT ..
861 PRINT ..
862 PRINT ..
863 PRINT ..
864 PRINT ..
865 PRINT ..
866 PRINT ..
867 PRINT ..
868 PRINT ..
869 PRINT ..
870 PRINT ..
871 PRINT ..
872 PRINT ..
873 PRINT ..
874 PRINT ..
875 PRINT ..
876 PRINT ..
877 PRINT ..
878 PRINT ..
879 PRINT ..
880 PRINT ..
881 PRINT ..
882 PRINT ..
883 PRINT ..
884 PRINT ..
885 PRINT ..
886 PRINT ..
887 PRINT ..
888 PRINT ..
889 PRINT ..
890 PRINT ..
891 PRINT ..
892 PRINT ..
893 PRINT ..
894 PRINT ..
895 PRINT ..
896 PRINT ..
897 PRINT ..
898 PRINT ..
899 PRINT ..
900 PRINT ..
901 PRINT ..
902 PRINT ..
903 PRINT ..
904 PRINT ..
905 PRINT ..
906 PRINT ..
907 PRINT ..
908 PRINT ..
909 PRINT ..
910 PRINT ..
911 PRINT ..
912 PRINT ..
913 PRINT ..
914 PRINT ..
915 PRINT ..
916 PRINT ..
917 PRINT ..
918 PRINT ..
919 PRINT ..
920 PRINT ..
921 PRINT ..
922 PRINT ..
923 PRINT ..
924 PRINT ..
925 PRINT ..
926 PRINT ..
927 PRINT ..
928 PRINT ..
929 PRINT ..
930 PRINT ..
931 PRINT ..
932 PRINT ..
933 PRINT ..
934 PRINT ..
935 PRINT ..
936 PRINT ..
937 PRINT ..
938 PRINT ..
939 PRINT ..
940 PRINT ..
941 PRINT ..
942 PRINT ..
943 PRINT ..
944 PRINT ..
945 PRINT ..
946 PRINT ..
947 PRINT ..
948 PRINT ..
949 PRINT ..
950 PRINT ..
951 PRINT ..
952 PRINT ..
953 PRINT ..
954 PRINT ..
955 PRINT ..
956 PRINT ..
957 PRINT ..
958 PRINT ..
959 PRINT ..
960 PRINT ..
961 PRINT ..
962 PRINT ..
963 PRINT ..
964 PRINT ..
965 PRINT ..
966 PRINT ..
967 PRINT ..
968 PRINT ..
969 PRINT ..
970 PRINT ..
971 PRINT ..
972 PRINT ..
973 PRINT ..
974 PRINT ..
975 PRINT ..
976 PRINT ..
977 PRINT ..
978 PRINT ..
979 PRINT ..
980 PRINT ..
981 PRINT ..
982 PRINT ..
983 PRINT ..
984 PRINT ..
985 PRINT ..
986 PRINT ..
987 PRINT ..
988 PRINT ..
989 PRINT ..
990 PRINT ..
991 PRINT ..
992 PRINT ..
993 PRINT ..
994 PRINT ..
995 PRINT ..
996 PRINT ..
997 PRINT ..
998 PRINT ..
999 PRINT ..
1000 PRINT ..

```

I dette program gælder det om at gætte den by på Danmarks kortet, som vises af ZX81. Først skal man indtaste hovedprogrammet. Derefter skrives GOTO 15, og alle byerne, incl. koordinater og data indtastes. Programmet er nu klar til brug. Hvis man ved et uheld kommer til at BREAK'e programmet, genopstartes

det ved GOTO 1. BRUG ALDRIG RUN!!!, da dette sletter alle data, og dermed masser af arbejde. Hvis du vil tillægge flere byer, kan variabelen N i linie 60 gøres større. Gøres variabelen O i linie 612 større, bliver man hørt flere gange og omvendt. Programmet SAVE's ved GOTO 9998. □

Thomas Nielsen

```

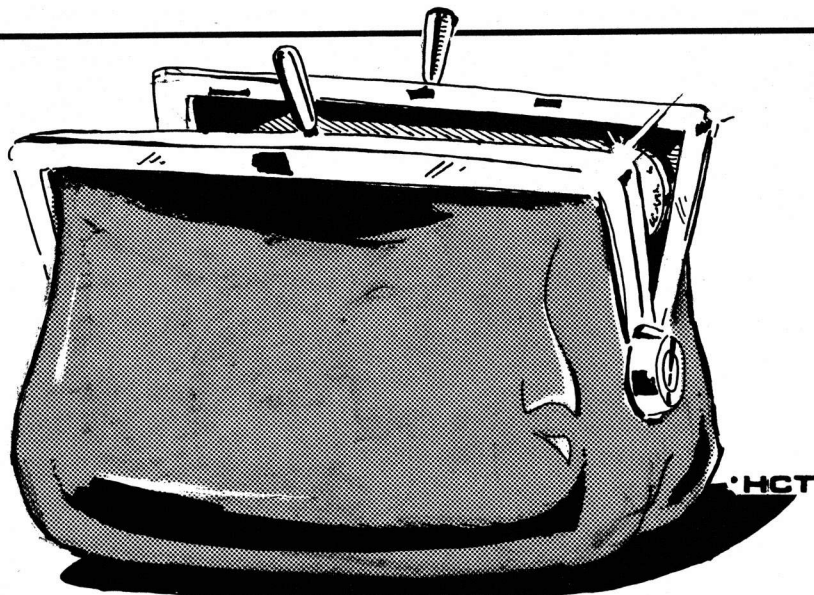
SKAGEN /57/40/ BYEN ER DA
NMARKS NORDLIGSTE BY HJØRRING
/34/37/ BYEN ER VEDSYESSEL 5
HOVEDSTAD FREDERIKSHAUN /37
/37/GØTEBORG - ..... /37/10/ HERFRA G
BRØNDESLÆU /34/35/ VAR I
1800 EN STOR MARKESBY AALB
ORG /34/32/ BYEN HAR DANMA
RKS ENESTE TUNNEL THISTED
/27/33/ DET ER THYS STØRSTE BY
LEMVIG /23/27/
BYENS SIDSTE STAVELSE ER NIG
STRUER /25/26/ B OG OS H
QUEDKVARTER LIGGER HER VIBORG
/30/26/ VAR I 1800 JYLLAND
S HOVEDSTAD HOBØ /3
4/30/ LIGGER INDERST I MARIAGER F
JORD RANDERS /34/26/ GUDE
NAREN UDLØBER I BYEN GRE
NAR /39/25/ HUNDESTED - .
..... EBELTOFT
0/38/23/ ODDEN - ..... /35/22
/ BYEN ER JYLLANDS STØRSTE BY
SILKEBORG 0/30/23/ S ENESTE
RACERBANE = ? DANMARKHERNING
/27/23/ DEN KALDES HANDESK
ERNES BY RINGKØBING /
23/23/1. STAVELSE = I BRYLUPSSER
MONI SKJERN /24/20/ MAD
S FRA KORSBAEK KOM HERFRA VE
JLE /32/17/ ALLAN SIMONS
EN BOR HER ESBJERG
/24/15/ DANMARKS STØRSTE HAVN
LIGGER I FREDERICIA /32/1
6/ DANMARKS MEST MODERNE BRYGGERI
= HADERSLÆU /30/11/ BYEN LI
GGER I SØNDERJYLLAND RIBE
/26/12/ DANMARKS ÆLDSTE
DOMKIRKE = ? TØNDER
/26/8/ DEN STØRSTE GRÆNSEBY = ..
..... SØNDERBORG /32/8/ HER
VAR CHRISTIAN DEN 2. FANGE MID
DEL FART 0/33/14/ DEN SIDSTE ST
AVELSE = FART ODENSE
/36/14/ HER LEVEDE EN BERØMT F
ORFATTER NYBORG /36/13
/KORSØR - .....
SVENDBORG /37/10/ HERFRA G
RAR EN BRO TIL TAASINGE RUDKØBI
NG /39/8/ DET ER LANGELANDS
STØRSTE BY NYKØBING SJ /4
5/19/ MORTEN FROST ER FØDT HER
KALUNDBORG /41/16/ DEN
FEMTÅRNED KIRKE LIGGER I HOL
BÆK /45/16/ BYEN LIGGER U
ED I SEFJORDEN FREDERIKSØRE
RK /47/19/ DANMARKS STØRSTE KRUDT
VÆRK = HELSINGØR /50/20
/HER LIGGER SLOTTET KRONBORG
HILLERØD /49/19/ HER LIGG
ER FREDENSBORG SLOT FREDERI
KSSUND /47/16/ HER SPILLES DER U
IKINGESPIL ROSKILDE /
47/16/ HER FINDES ET VIKINGEMUSEU
M KØBENHAVN /50/17/ DRØ
NNINGEN BOR I DENNE STØRE BY SL
AGELSE /43/14/ HER LIGGER S
T. PEDERS KIRKE RINGSTED
/46/14/ BYEN ER EN GAMMEL VIK
INGEBY KØGE /48/1
4/ BYEN LIGGER NÆR KØBENHAVN
NÆSTVED /46/11/ HER LUF
S HOLM KOSTSKOLE LIGGER I KORSØR
/41/13/ NYBORG - .....
VORDBJERG
/46/9/ HER LIGGER GÅRSTÅRNET
STEGE /49/8/ BYEN
LIGGER PÅ MØN NYK
ØBING F /46/6/ DET ER FALSTER
S HOVEDSTAD RØDEY
/44/5/ PUTTGÅRDEN - .....
NAKSKOV /41/6/ HE
R LIGGER ET DM TALT SKIBSVAERFT R
ØNNE /55/4/ DEN LIGGER P
RÅ BORNHOLM

```

```

450 PRINT " "
460 PRINT " "
470 PRINT " "
480 PRINT " "
490 PRINT " "
510 FOR N=0 TO 52
520 PLOT N,42
530 PLOT N,1
540 NEXT N
550 FOR N=1 TO 42
560 PLOT 0,N
570 PLOT 1,N
580 PLOT 52,N
590 NEXT N
600 REM "PRIKKNING AF BYER"
610 LET C=INT (RND*50)+1
620 IF C=50 THEN GOTO 1500
630 IF A$(C,15)="0" THEN GOTO 6
1000
640 FOR N=0 TO 10
650 PLOT A(C),B(C)
660 FOR M=0 TO 5
670 NEXT M
680 UNPLOT A(C),B(C)
690 NEXT N
700 LET C=C+1
710 LET J=0
720 PRINT AT 2,1;"BYNAVN ?"
730 INPUT C$(1)
740 IF C$(1) > A$(C) THEN GOTO 3
40
750 PRINT AT 2,1;"RISTIGT ";AT
3,1;"DET VAR"
760 FOR N=5 TO 10
770 PRINT AT N,4;A$(C,N-4)
780 NEXT N
790 IF J=0 THEN LET P=P+10
800 IF J=1 THEN LET P=P+5
810 LET A$(C,15)="0"
820 PRINT AT 19,13;P
830 PLOT A(C),B(C)
840 GOSUB 1200
850 GOTO 600
860 PRINT AT 2,1;"
870 IF J=1 THEN GOTO 1300
880 LET J=1
890 LET U$="DESVAERRE MEN PAA"
900 PRINT AT 10,1;U$(1 TO 3)
910 LET U$=U$(2 TO 3)+U$(1)
920 IF INKEY$="0" THEN GOTO 910
930 GOTO 870
940 FOR N=1 TO 20
950 PRINT AT N,0;" "
960 NEXT N
970 PRINT AT 2,2;"TAST 0"
980 PRINT AT 10,1;B$(C,1 TO 3)
990 LET B$(C)=B$(C,2 TO 3)+B$(C,
1)
970 IF INKEY$="0" THEN GOTO 990
980 GOTO 950
990 GOSUB 1200
1000 GOTO 700
1200 FOR N=1 TO 20
1210 PRINT AT N,0;" "
1220 NEXT N
1230 RETURN
1300 PRINT AT 2,1;"
1310 LET U$="NEJ DESVAERRE DET U
RR "+A$(C)
1320 FOR N=1 TO 20
1330 PRINT AT N,0;" "
1340 NEXT N
1345 FOR N=0 TO 94
1350 PRINT AT 10,1;U$(1 TO 3)
1360 LET U$=U$(2 TO 3)+U$(1)
1370 NEXT N
1375 PLOT A(C),B(C)
1380 GOSUB 1200
1390 LET A$(C,15)="0"
1400 GOTO 600
1500 PRINT AT 0,0;"
*****
*****
*****
1500 FOR N=3 TO 19
1510 PRINT AT N,0;" "
1520 NEXT N
1525 PRINT AT 19,0;"
*****
*****
*****
1530 PRINT AT 5,7;"SAA SLUTTER U
DRES";AT 7,7;"DATABEGRAFITINE"
1532 IF P=500 THEN LET O$="TRETT
EN UOPNAEELIGT"
1534 IF P<500 AND P>460 THEN LET
O$="ELLVE SHUKT"
1536 IF P<460 AND P>400 THEN LE
T O$="TI PERFECT"
1538 IF P<400 AND P>320 THEN LE
T O$="NI GODT"
1540 IF P<320 AND P>270 THEN LE
T O$="OTTE RIMELIGT"
1542 IF P<270 AND P>200 THEN LE
T O$="SYU NOENLUNDE"
1544 IF P<200 AND P>150 THEN LE
T O$="SEKS DAARLIGT"
1546 IF P<150 AND P>80 THEN LET
O$="FEM SINDSOPRIVENDE"
1548 IF P<80 AND P>0 THEN LET O
$="TRE PRUO ATLAS"
1550 IF P=0 THEN LET O$="MUL
FRYDELIGT"
1555 PRINT AT 9,7;"DU FIK ";O$(1
TO 7)
1570 PRINT AT 10,7;O$(8 TO 1)
1575 PRINT AT 13,6;"TAST 0 FOR N
YT SPIL";AT 14,6;"TAST 5 FOR AT
STOPPE"
1590 IF INKEY$="5" THEN STOP
1595 IF INKEY$="0" THEN GOTO 160
0
1600 GOTO 1500
1605 FOR N=1 TO 50
1610 LET A$(N,15)=" "
1620 NEXT N
1630 CLS
1640 GOTO 1
1650 FOR N=1 TO 50
1660 PRINT B$(N)
1670 NEXT N
1680 STOP
1690 SAVE "GE"
1700 GOTO 1

```



Budget

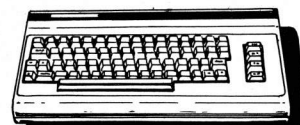
I dette lille, men smarte bud-
getprogram har du mulighed
for at indtaste forskellige ud-
gifter og dernæst få din Com-
modore 64 til at lægge talle-
ne sammen og komme med
et flot skema. □ *F. Trap*

```

5 DIMB$(15),B(15)
10 PRINT" I FØRSTE KOLONNE SKRIVES ART"
20 PRINT" I ANDEN KOLONNE SKRIVES KR."
30 PRINT" TRYK RETURN"
40 GETA$:IFA$<>CHR$(13)THEN 40
50 PRINT" BUDGET : "
60 PRINT"ART",,"KR.,""IALT"
70 PRINT"-----"
80 FOR T=1 TO 15
90 INPUTB$(T)
100 PRINT" ",,;INPUTB(T)
110 IA=IA+B(T)
120 PRINT" ",,;IA
130 NEXTT
140 PRINT"-----"
150 PRINT" IALT",,;IA
160 GETA$:IFA$<>CHR$(13)THEN160
170 PRINT" BUDGET : "
180 PRINT"NR ART",,"KR.,""IALT"
190 PRINT"-----"
200 FOR T=1 TO 15
210 PRINT T;B$(T)
220 PRINT" ",,;B(T)
230 IS=IS+B(T)
240 PRINT" ",,;IS
250 NEXTT
260 PRINT"-----"
270 PRINT" IALT",,;IS
280 GETA$:IFA$<>CHR$(13)THEN 280
290 PRINT" HVILKET NR. ØNSKES RENDRET "
300 PRINT" (HVIS IKKE TASTES 0)"
310 INPUT"NR. ";NR
320 IFNR=0THEN 170
330 PRINT"ART",,"KR."
340 PRINT" ",B$(NR),,;B(NR)
350 INPUT" ";B$(NR)
360 PRINT" ",,;INPUTB(NR);IS=0
370 GOTO 170

```

COMMO-
DORE
64



Mastermind



Mastermind hører til markedets mest populære spil. I denne udgave passer det til Commodore 64.

Spillet er opbygget, så computeren selv bestemmer, om der skal gættes fem eller seks

farver, - naturligvis alt efter, hvor dygtig du er. Bemærk i øvrigt, at farverne i den skjul-

te kombination er ikke nødvendigvis forskellige.

L. M. Laursen

```

20 POKE53280,11:POKE53281,11:PRINT"COMMO-DORE 64"
30 PRINT"MASTERMIND"
40 PRINT"REGLER J-N?"
50 GETA$:IFA$=""THEN50
60 IFA$="N"THENZ=6:GOTO170
70 PRINT"COMPUTEREN UDFYLDER 4 FELTER MED TILFÆLDIGE FARVER."
80 PRINT"EKSEMPEL: N O P Q"
90 PRINT"KOMBINATIONEN SKAL DU SAA GAETTE."
100 PRINT"O: VISER DIG HVOR MANGE FARVER DU HAR PLACERET RIGTIGT."
110 PRINT"O: VISER DIG HVOR MANGE FARVER DER ER RIGTIGE."
120 PRINT"MEN SIDDER FORKERT."
130 PRINT"O: TRYK PAA N: J."
140 GETA$:IFA$="J"THEN160
150 GOTO140
160 Z=5
170 S=1
180 D=0
190 L=1
199 REM*****
200 REM OPRETTE SKAERMILLEDE
210 PRINT"MASTERMIND"
220 FORX=0TO24
230 FORY=1TO9
240 POKE1024+20+Y+(X*40),160:POKE55296+20+Y+(X*40),0
250 NEXTY:NEXTX
260 IFZ=6THENPRINT"NY CYAN=4:IT"
270 PRINT"NY HVID=2":PRINT"NY RØD=3":PRINT"NY GRØN=5":PRINT"NY BLÅ=7":PRINT"NY GUL=8"
280 PRINT"NY RIGTIG":PRINT"PLACERE!"
290 PRINT"NY RIGTIG":PRINT"FARVE"
299 REM*****
300 REM GENERERE FARVER
310 FORX=1TO4
320 F(X)=INT(Z*RND(1))+9-Z
330 IFF(X)=4THENF(X)=2
340 IFF(X)=5THENF(X)=3
350 NEXT
399 REM*****
400 REM INPUT
410 PRINT"NY FØRSTE FARVE ?":X=1:GOSUB500
420 PRINT"NY ANDEN FARVE ?":X=2:GOSUB500
430 PRINT"NY TREDIE FARVE ?":X=3:GOSUB500
440 PRINT"NY FJERDE FARVE ?":X=4:GOSUB500
450 GOTO700
500 GETA$:IFA$=""THEN500
510 IFA$="2"THENG(X)=2:GOTO600
520 IFA$="3"THENG(X)=3:GOTO600
530 IFA$="6"THENG(X)=6:GOTO600
540 IFA$="7"THENG(X)=7:GOTO600
550 IFA$="8"THENG(X)=8:GOTO600
560 IFZ=6ANDIFA$="4"THENG(X)=4:GOTO600
570 GOTO500
600 POKE55316+(X*2)+(L*40),G(X)-1
610 RETURN
700 PRINT"ER DU SIKKER? J-N"
710 GETA$:IFA$=""THEN710
720 PRINT"NY"
730 IFA$="N"THENPRINT"IT":GOTO400
740 IFA$="J"THENPRINT"IT":GOTO800
750 GOTO700
800 REM SAMMENLIGNING
810 P=0:F=0
820 FORX=1TO4
830 IFF(X)=G(X)THENP=P+1:F(X)=F(X)*10:G(X)=G(X)*100
840 NEXT
900 FORX=1TO4
910 FORY=1TO4
920 IFX=YTHEN940
930 IFF(X)=G(Y)THENF=F+1:F(X)=F(X)*10:G(Y)=G(Y)*100
940 NEXTY
950 NEXTX
960 FORX=1TO4
970 IFG(X)>100THENG(X)=G(X)/100
980 IFF(X)>10THENF(X)=F(X)/10
990 NEXT
999 REM*****
1000 REM POINTGIVNING
1010 POKE1055+(L*40),P+48:POKE1057+(L*40),F+48
1020 FORX=1TO3:POKE55326+X+(L*40),1:NEXT
1030 IFP=4THENFORX=1TO800:NEXT:GOTO1200
1099 REM*****
1100 REM NÆSTE GAET
1110 L=L+2
1120 IF(L+1)/2=13THENFORX=1TO800:NEXT:GOTO1200
1130 GOTO400
1199 REM*****
1200 REM ORIENTERING
1210 PRINT"NY"
1220 L=(L+1)/2:D=D+L:R=D/S
1230 IFL=13THENPRINT"NY TAK, LUDO ER NOK MERE DIN STIL":S=1:D=0:GOTO1300
1240 PRINT"NY DU KLAREDE DET I NY FORSØG."
1250 IFS>1THENPRINT"NY DIT GENNEMSNIET ER INT(R*100)/100":FORSØG
1260 S=S+1
1299 REM*****
1300 REM NYT SPIL?
1310 PRINT"NY VIL DU PRØVE IGEN? J-N"
1320 GETA$:IFA$=""THEN1320
1330 IFA$="J"THEN1400
1340 IFA$="N"THEN10
1350 GOTO1320
1399 REM*****
1400 REM SVARHEDSGRAD
1410 IFZ=6THEN1450
1420 IFL>7ORZ<4THENGOTO190
1430 PRINT"NY PAA PAA, NU FAAR DU EN FARVE TIL."
1440 Z=6:FORX=1TO4000:NEXT:GOTO190
1450 IFL=13THENPRINT"NY OK, MEN KUN MED FEM FARVER":FORX=1TO4000:NEXT
1460 IFL=13THENZ=5
1470 GOTO190
    
```

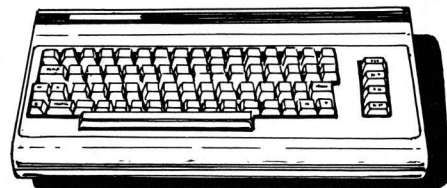

Cosmic Killer

COMMO-
DORE
64

Cosmic Killer er et Space Invader-spil, med sprites, farver og lyd. Når spillet indtastes skal du være omhyggelig med DATA-sætningerne, da

det er maskinkode data. Spillet, der kører på en CBM-64, er selvforklarende.

Niels Lavsten



Ø REM LAVET AF NIELS LAUSTEN

```
1 DATA 24,173,12,64,185,1,141,12,64,205,13,64,208,48,234,174,10,64,234
2 DATA 224,246,208,2,162,0,189,0,65,141,0,212,232,189,0,65,141,1
3 DATA 212,232,189,0,65,141,13,64,169,0,141,12,64,232,142,10,64,76,49,234
10 FORI=814*6+256T0814*6+256+56:READA:POKEI,A:NEXT
11 REM
12 REM MUSIK-MASKINKODE
13 REM
19 POKE814*4+14,1:POKE814*4+15,0
20 POKE54272,0:POKE54273,0:AN=3
21 POKE 54296,15
22 POKE 54277,255
23 POKE 54278,248
24 POKE54276,17
26 POKE814*4+10,00:POKE814*4+11,65
27 POKE814*4+12,0:POKE814*4+13,1
28 DATA 154,21,15,100,20,15,154,21,15,100,20,15,154,21,15,47,16,15
29 DATA 63,19,15,37,17,15,107,14,30
30 DATA 147,8,15,205,10,15,107,14,15,47,16,30,205,10,15,156,13,15,47,16,15
31 DATA 37,17,30,205,10,15,154,21,15,100,20,15
32 DATA 154,21,15,100,20,15,154,21,15,47,16,15
33 DATA 63,19,15,37,17,15
34 DATA 107,14,30,147,8,15,205,10,15,107,14,15,47,16,30,159,9,15
35 DATA 37,17,15,47,16,15,107,14,15
36 DATA 154,21,15,100,20,15,154,21,15,100,20,15,154,21,15,47,16,15
37 DATA 63,19,15,37,17,15,107,14,30
38 DATA 147,8,15,205,10,15,107,14,15,47,16,30,205,10,15,156,13,15,47,16,15
39 DATA 37,17,30,205,10,15,154,21,15,100,20,15
40 DATA 154,21,15,100,20,15,154,21,15,47,16,15
41 DATA 63,19,15,37,17,15
42 DATA 107,14,30,147,8,15,205,10,15,107,14,15,47,16,30,159,9,15
43 DATA 37,17,15,47,16,15,107,14,15
44 DATA 47,16,15,37,17,15,63,19,15,154,21,30,216,12,15,227,22,15,154,21,15
45 DATA63,19,30,114,1,15,154,21,15,63,19,15,37,17,30,0
46 FORI=16640T016640+246:READA:POKEI,A:NEXT
47 REM
48 REM MELODI-DATA
49 REM
67 DATA 160,0,185,112,7,201,32,240,14,201,96,240,10,201,27,144,6,169,1
68 DATA 141,6,64,96,200,192,48,144,230,169,0,141,6,64,96
70 FORI=24736T024736+33:READA:POKEI,A:NEXT
71 REM
72 REM 4 NEDERSTE RAEKKE
73 REM
75 DATA 172,2,64,169,192,133,251,169,7
77 DATA 133,252,234,234,234,177,251,201
79 DATA 32,240,48,201,96,240,44,201,27
81 DATA 144,40,234,234,234,24,165,251,105,40,133,251,144,8,24,165,252
83 DATA 105,1,133,252,234,169,0,145,251
85 DATA 24,165,252,105,212,133,252,173,5,64,145,251,76,93,96,234,234,56
87 DATA 165,251,233,40,133,251,176,7,56,165,252,233,1,133,252,234,165
89 DATA 252,201,3,208,177,234,234,152,24,109,4,64,168,204,3,64,144,154,96
90 FORI=24576T024576+105:READA:POKEI,A:NEXT
91 REM
92 REM SPACE.1. BOMBER
93 REM
100 DATA 169,0,141,1,64,169,192,133,251
110 DATA 169,7,133,252,160,0,234,177,251,201,0,240,18,201,27,144,14,201
120 DATA 32,240,10,201,96,240,6,169,1,141,1,64,96,234,192,0,208,5,160
130 DATA 39,76,11,89,234,56,165,251,233,40,133,251,176,8,56,165,252,233
140 DATA 1,133,252,234,165,252,201,3,208,195,96
150 FORI=22779T022779+74:READA:POKEI,A:NEXT
160 REM
170 REM UNDERSØG SIDERNE
180 REM
190 DATA 169,4,133,252,133,254,169,1,133,251,169,0,133,253,160,255,200
200 DATA 177,251,201,27,144,30,201,32,240,26,145,253,169,32,145,251,24,165
210 DATA 252,72,105,212,133,252,133,254,177,251,145,253,104,133,252,133
220 DATA 254,234,192,255,208,12,24,165
230 DATA 252,105,1,133,252,133,254,76,16,85,192,231,208,199,165,252,201
240 DATA 7,208,193,96
250 FORI=814*5+256*5T0814*5+256*5+79:READA:POKEI,A:NEXT:REM FLYT TIL VENSTRE
260 REM
270 REM
280 DATA 162,3,202,173,0,64,201,1,240,103,173,1,208,201,255,208,48,234
290 DATA 165,197,201,10,208,89,234,173,16,208,74,144,12,173,16,208,9
300 DATA 1,141,16,208,234,76,72,88,173,16,208,41,254,141,16,208,234,173
310 DATA 2,208,141,0,208,173,3,208,141
320 DATA 1,208,234,234,173,1,208,136,140,1,208,234,234,234,192,10,208,14
330 DATA 169,255,141,1,208,169,0,141,0,64,76,133,88,234,173,31,208,74
340 DATA 176,7,224,0,208,153,76,133,88
350 DATA 169,1,141,0,64,76,00,97
360 FORI=22548T022663:READA:POKEI,A:NEXT
370 REM BOMBE
380 REM
390 DATA 162,1,202,165,197,201,44,240,8
400 DATA 201,47,240,57,76,136,86,234,173,2,208,24,105,1,176,27,201,64
410 DATA 240,7,141,2,208,76,136,86,234,173,16,208,74,74,176,79,169,64,141
420 DATA 2,208,76,136,86,234,173,16,208,9,2,141,16,208,169,0,141,2
430 DATA 208,76,136,86,234,234,56,173,2,208,233,1,144,27,201,24,240,7,141
440 DATA 2,208,76,136,86,234,173,16,208,74,74,144,26,169,24,141,2,208
450 DATA 76,136,86,234,173,16,208,56,233,2,141,16,208,169,255,141,2,208
460 DATA 76,136,86,224,0,208,132,76,20,88
470 FORI=22030T022030+128:READA:POKEI,A:NEXT:REM FLYT RUMSKIB
480 REM
490 REM
500 DATA 169,192,133,251,169,6,133,252,169,232,133,253,169,6,133,254,160
510 DATA 0,234,192,63,208,7,165,252,201,3,208,1,96,234,136,177,251,201
```

```
520 DATA 27,144,31,201,32,240,27,234,234,145,253,169,32,145,251,165,252
530 DATA 72,24,105,212,133,252,133,254,177,251,145,253,104,133,252,133,254
540 DATA 234,192,0,208,200,165,252,234,233,1,133,252,133,254,76,18,83
550 DATA 0,73,0,143,38,182
560 FORI=21248T021248+91:READA:POKEI,A:NEXT:REM FLYT SPACEI NED
570 PRINT"XXXXXXXXXX"X:JEJRE ">";VENSTRE "<";SKYD "A"
580 POKE52,48:POKE56,48:POKE53281,15:POKE53280,6
590 DATA 160,0,169,32,153,192,7,200,192
600 DATA 41,208,248,234,169,191,133,251,169,7,133,252,160,0,177,251,201
610 REM
620 DATA 0,208,48,234,234,169,32,145,251,160,40,177,251,201,32,208,29,234
630 DATA 169,0,145,251,24,165,252,105
640 DATA 212,133,252,160,0,177,251,160,40,145,251,56,165,252,233,212,133
650 DATA 252,234,234,201,96,240,223,234
652 DATA 234,165,251,56,233,1,133,251,176,190,234,56,165,252,233,1,133
654 DATA 252,201,3,208,178,96,35,82,96
660 FORI=20992T020992+102:READA:POKEI,A:NEXT
670 REM FLYT S.I. BOMBER NED
680 REM
690 POKE52,48:POKE56,48
700 FORI=0T097:READA:POKEI+814*5,A:NEXT
710 DATA 160,0,24,169,48,133,52,169,48,133,56,173,14,220,41,254,141,14
720 DATA 220,165,1,41,251,133,1,162,0,189,0,208,157,0,48,232,224,0,208
730 DATA245,200,173,29,80,24,105,1,141,29,80,173,32,80,105,1,141,32
740 DATA 80,192,16,208,223,165,1,9,4,133
750 DATA 1,173,14,220,9,1,141,14,220,173,24,208,41,240,24,105,12,141,24
760 DATA 208,169,0,169,208,141,29,80,169,48,141,32,80,96
770 SYS 814*5:REM FLYT TEGN TIL RAM
810 DATA169,175,141,217,48
820 DATA169,175,141,218,48
830 DATA169,16,141,221,48
840 DATA169,4,141,222,48
850 DATA169,1,141,223,48
860 DATA169,175,141,233,48
870 DATA169,175,141,234,48
880 DATA169,16,141,237,48
890 DATA169,4,141,238,48
900 DATA169,1,141,239,48,96
910 FORI=832T0882:READA:POKEI,A:NEXT
920 DATA169,250,141,217,48
930 DATA169,250,141,218,48
940 DATA169,4,141,221,48
950 DATA169,16,141,222,48
960 DATA169,64,141,223,48
970 DATA169,250,141,233,48
980 DATA169,250,141,234,48
990 DATA169,4,141,237,48
1000 DATA169,16,141,238,48
1010 DATA169,64,141,239,48,96
1020 FORI=896T0946:READA:POKEI,A:NEXT
1030 DATA169,4,141,0,48
1040 DATA169,4,141,1,48
1050 DATA169,12,141,2,48
1060 DATA169,12,141,3,48
1070 DATA169,12,141,4,48
1080 DATA169,12,141,5,48
1090 DATA169,4,141,6,48
1100 DATA169,4,141,7,48,96
1110 FORI=49152T049152+40:READA:POKEI,A:NEXT
1120 DATA169,1,141,0,48
1130 DATA169,1,141,1,48
1140 DATA169,12,141,2,48
1150 DATA169,12,141,3,48
1160 DATA169,48,141,4,48
1170 DATA169,48,141,5,48
1180 DATA169,64,141,6,48
1190 DATA169,64,141,7,48,96
1200 FORI=49193T049193+40:READA:POKEI,A:NEXT
1220 DATA169,0,141,0,48
1230 DATA169,0,141,1,48
1240 DATA169,0,141,2,48
1250 DATA169,125,141,3,48
1260 DATA169,125,141,4,48
1270 DATA169,0,141,5,48
1280 DATA169,0,141,6,48
1290 DATA169,0,141,7,48,96
1300 FORI=49234T049234+40:READA:POKEI,A:NEXT
1310 DATA169,64,141,0,48
1320 DATA169,64,141,1,48
1330 DATA169,48,141,2,48
1340 DATA169,48,141,3,48
1350 DATA169,12,141,4,48
1360 DATA169,12,141,5,48
1370 DATA169,1,141,6,48
1380 DATA169,1,141,7,48,96
1390 FORI=49275T049275+40:READA:POKEI,A:NEXT
1400 A=1024+38:X=152:Y=53248:W=1024:Y=229
1410 POKE53270,PEEK(53270)OR16
1420 POKE53282,2:POKE53283,0
1430 FORI=0T07:READA:POKE27*8+I+12288,A:NEXT
1440 DATA63,175,175,255,21,4,16,64
1450 FORI=0T07:READA:POKE29*8+I+12288,A:NEXT
1460 DATA252,175,175,255,84,4,16,64
1470 REMSTOP
1480 FORI=254*64T0254*64+63:POKEI,0:NEXT
1490 FORI=254*64+1T020+254*64STEP3:POKEI,32:NEXT
1500 FORI=11*64T011*64+63:READA:POKEI,A:NEXT
```


Juveler

Forbryderen "No Good" har i løbet af sit lange liv røvet op til flere juveler, som han har gemt i flere etagers grotte. Detektiven (dig) har fundet grotten, og ved hjælp af en en-mands-jet forsøger du at få fat i juvelerne før tiden løber ud. □

Karsten Knoblauch



```

1 BORDER 0: PAPER 0: INK 7: BRIGHT 1:
CLS
6 LET h=2000
10 FOR f=0 TO 50: BEEP .005,f+(10*RND)
: NEXT f
20 FOR j=2 TO 7: PRINT AT 10,3: RESTO
RE 20: FOR f=0 TO 26: READ r: PRINT INK
j;CHR$ r: BEEP .01,f: NEXT f
30 DATA 127,128,75,65,82,83,84,69,78,1
28,75,78,79,66,76,65,85,67,72,128,45,128
,49,57,56,52,46
40 NEXT j
50 INPUT "SVAERHEDSGRAD(0-20)";o
55 IF o>20 OR o<0 THEN GO TO 50
60 CLS
70 LET .sc=0
80 LET l=3
100 LET b=15
101 LET q=18
102 LET t=500
103 LET d=0
104 LET a=0
500 REM
501 REM ~q~ = GRAPH. ~q~
502 REM
503 REM ~e~ = GRAPH. ~e~
504 REM
505 REM ~r~ = GRAPH. ~r~
506 REM
507 REM ~t~ = GRAPH. ~t~
508 REM
509 REM ~u~ = GRAPH. ~u~
510 REM
511 REM ~i~ = GRAPH. ~i~
512 REM
513 REM ~o~ = GRAPH. ~o~
520 REM
600 IF o<10 THEN RESTORE 600: FOR n=0
TO 7: READ r: POKE USR "o"+n,r: NEXT n
700 DATA 0,0,0,124,254,124,56,16
1000 RESTORE 1000: FOR j=1 TO 16: READ r
: READ s: READ u:: FOR f=r TO s: PRINT
INK 4;AT u,f;"q": BEEP .005,f: NEXT f: N
EXT j
1010 DATA 5,15,5,20,30,5,0,5,7,10,20,7,0
,15,9,20,30,9,0,5,11,10
1020 DATA 20,11,25,30,11,0,5,13,15,30,13
,3,20,15,25,30,15,0,10
1030 DATA 17,15,25,17,0,30,19
1200 FOR f=1 TO 29: PRINT AT 0,f: INK 6;
"e": BEEP .005,f: NEXT f
1210 PRINT AT 1,0: INK 6;"uSCORE:0 L
IV:|||| TID: t"
1300 FOR f=1 TO 29: PRINT AT 2,f: INK 6;
"r": BEEP .005,f: NEXT f
1310 FOR f=1 TO 29: PRINT AT 20,f: INK 6
;"e": BEEP .005,f: NEXT f
1360 PRINT INK 6;AT 21,0;"u": FOR f=1 T
O 29: PRINT AT 21,f; INK 6;" ": BEEP .00
5,f: NEXT f: PRINT INK 6;AT 21,30;"t"
1380 PRINT AT 21,1: INVERSE 1: INK 6;"OM
GANG: HI: o:"
1399 INK 7
1400 FOR f=0 TO 6: READ r: READ s: READ
u: PRINT INK (RND*5);AT r,(RND*5)+u;"o"
: LET a=a+1: BEEP .01,f*2: NEXT f
1410 DATA 4,8,6,6,8,11,8,13,1,10,8,11,12
,3,1,14,8,11,16,8,1
1500 IF o>1 THEN PRINT INK (RND*5);AT
18,(RND*12)+1;"o": LET a=a+1
1510 IF o>2 THEN PRINT INK (RND*5);AT
18,(RND*12)+17;"o": LET a=a+1
1520 IF o>3 THEN PRINT INK (RND*5);AT

```



```

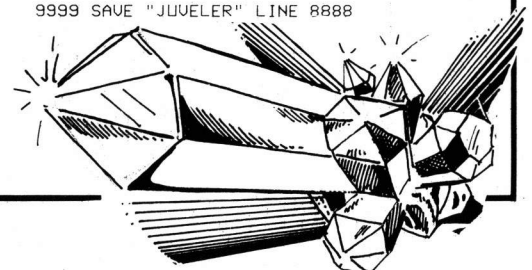
16,(RND*8)+16;"o": LET a=a+1
1530 IF o>4 THEN PRINT INK (RND*5);AT
12,(RND*13)+16;"o": LET a=a+1
1540 IF o>5 THEN PRINT INK (RND*5);AT
8,(RND*8)+21;"o": LET a=a+1
1550 IF o>6 THEN PRINT INK (RND*5);AT
4,(RND*8)+21;"o": LET a=a+1
1560 IF o>7 THEN PRINT INK (RND*5);AT
6,(RND*3)+1;"o": LET a=a+1
1570 IF o>8 THEN PRINT INK (RND*3);AT
10,(RND*3)+1;"o": LET a=a+1
1580 IF o>9 THEN PRINT INK (RND*3);AT
10,(RND*3)+26;"o": LET a=a+1
1590 IF o>10 THEN PRINT INK (RND*3);AT
14,(RND*3)+26;"o": LET a=a+1
1600 IF o>11 THEN LET t=t-(o*10)
1610 IF t<300 THEN LET t=300
1620 IF o>8 THEN RESTORE 1600: FOR n=0
TO 7: READ r: POKE USR "o"+n,r: NEXT n
1650 DATA 0,146,84,56,254,56,84,146
1900 PRINT INVERSE 1: INK 6;AT 21,15;h
1980 PRINT INVERSE 1: INK 6;AT 1,17+l;"
"
2000 PRINT INK 6: INVERSE 1;AT j,26;t;"
"
2010 LET t=t-1.
2500 PRINT INVERSE 1: INK 6;AT 21,8;o
2510 PRINT INVERSE 1: INK 6;AT 1,7;sc
2520 PRINT INVERSE 1: INK 6;AT 21,26;a-
o;" "
2999 PRINT AT q,b;" "
3000 LET b=b+(INKEY$="3")-(INKEY$="1")
3010 IF b<0 THEN LET b=30
3011 IF b>30 THEN LET b=0
3100 IF SCREEN$ (q+1,b)=" " AND INKEY$<
"0" THEN LET q=q+1
3200 IF SCREEN$ (q,b)<>" " THEN BEEP .0
1,q+b*10: LET d=d+1: LET sc=sc+25
3300 IF t=0 THEN LET l=l-1:: FOR f=69
TO -60 STEP -1: BEEP .01,f: NEXT f: IF l
>0 THEN CLS: GO TO 100
3310: IF l=0 THEN CLS: GO TO 7000
3400 IF d=a THEN LET o=o+1: LET sc=sc+1
00: CLS: GO TO 100
3990 IF SCREEN$ (q-1,b)=" " AND INKEY$="
0" THEN LET q=q-1
4000 PRINT AT q,b;"i"
6000 GO TO 2000
7000 LET a$="SLUT OPGØR"
7010 LET b$=" "
7050 IF sc>h THEN LET h=sc
7100 FOR f=10 TO 20: PRINT INK (f-10)/2
;AT 10,10;a$; OVER 1;AT 10,10;b$: BEEP .
005,f: NEXT f
7150 PRINT AT 15,0
7200 PRINT "SCORE =",sc
7210 PRINT "TID =",t
7220 PRINT "OMGANG=",o
7230 PRINT "o" =,"a-d
7300 PRINT #1;" VIL DU PRØVE IGEN ? (J
/N/M) "
7400 IF INKEY$="n" THEN GO TO 7450
7401 IF INKEY$="j" THEN GO TO 7500
7402 IF INKEY$="m" THEN GO TO 7600
7410 BEEP .01,RND*60
7420 GO TO 7400
7450 CLS: PRINT INVERSE 1;AT 11,0;" D
U VILLE GERNE SLUTTE SPILLET. ": BEEP .5
,10: STOP
7500 CLS: PRINT INVERSE 1;AT 11,0;"
DU VILLE GERNE SPILLE IGEN. ": BEEP .5
,30: CLS: GO TO 10
7600 CLS: PRINT INVERSE 1;AT 11,0;"

```

```

DU VILLE MÅSKE SPILLE IGEN "
7610 PRINT INVERSE 1;AT 12,0;" SAA JEG
TRAF VALGET FOR DIG..."
7620 IF (RND*10)<5 THEN LET d$="IKKE "
: GO TO 7622
7621 LET d$=""
7660 PRINT INVERSE 1;AT 15,0;" DU VILL
E ";d$;"SPILLE IGEN..."
7700 IF d$="IKKE " THEN BEEP .5,10: STO
P
7710 IF d$="" THEN BEEP 2.5,30: CLS: G
O TO 10
8000 STOP
8888 RESTORE 9000
8900 BORDER 0: PAPER 0: INK 7: BRIGHT 1:
CLS
9000 FOR n=0 TO 7: READ r: BEEP .01,n: P
OKE USR "q"+n,r: NEXT n
9010 DATA 0,126,254,255,255,239,247,102
9020 FOR n=0 TO 7: READ r: BEEP .01,n: P
OKE USR "e"+n,r: NEXT n
9030 DATA 0,0,0,0,16,116,254,255
9040 FOR n=0 TO 7: READ r: BEEP .01,n: P
OKE USR "r"+n,r: NEXT n
9050 DATA 255,254,116,16,0,0,0,0
9060 FOR n=0 TO 7: READ r: BEEP .01,n: P
OKE USR "t"+n,r: NEXT n
9070 DATA 192,224,224,240,192,224,192,12
8
9080 FOR n=0 TO 7: READ r: BEEP .01,n: P
OKE USR "u"+n,r: NEXT n
9090 DATA 3,7,7,15,3,7,3,1
9100 FOR n=0 TO 7: READ r: BEEP .01,n: P
OKE USR "i"+n,r: NEXT n
9110 DATA 24,60,24,60,90,24,36,102
9120 FOR n=0 TO 7: READ r: BEEP .01,n: P
OKE USR "o"+n,r: NEXT n
9130 DATA 0,0,0,124,254,124,56,16
9300 FOR f=0 TO 7: PRINT INK (RND*6)+1
;"JUVELER "; BEEP .01,RND*60: NEXT f
9400 CLS: PRINT FLASH 1;AT 0,10;"INSTR
UKTIONER"
9410 PRINT AT 2,0;"FORBRYDEREN ~NOGOOD~
HAR I LØBETAF SIT LANGE LIV RØVET OP TIL
FLERE JUVELER SOM HAN HAR GEMT I EN
FLER-ETAGES GROTTE."
9420 PRINT AT 7,0;"DETEKTIVEN (dig) HAR
FUNDET GROTTEN ,OG VED HJÆLP AF EN
EN-MANDS-JET FORSØGER HAN AT FAAFAT I
JUVELERNE FØR TIDEN LØBER UD....."
9430 PRINT INVERSE 1;AT 12,0;" NAAR
MAN KOMMER TIL CA. OMGANG NR.9
, KAN DER FOREKOMME ~SORTE~
JUVELER
(DE KAN IKKE SES.)
9450 PRINT " DU STYRER MED:"
9460 PRINT : PRINT INVERSE 1;" 1=UENSTR
E 3=HØJRE 0=OP "
9490 LET a=RND*30
9500 PRINT INK (a/5)+1: INVERSE 1;AT 21
,0;" TAST ~P~ FOR AT SPILLE JUVELER "
9551 LET a=a+10
9560 BEEP .01,a: BEEP .005,a
9600 IF INKEY$<"p" THEN GO TO 9490
9700 RUN
9999 SAVE "JUVELER" LINE 8888

```



Q-Bert

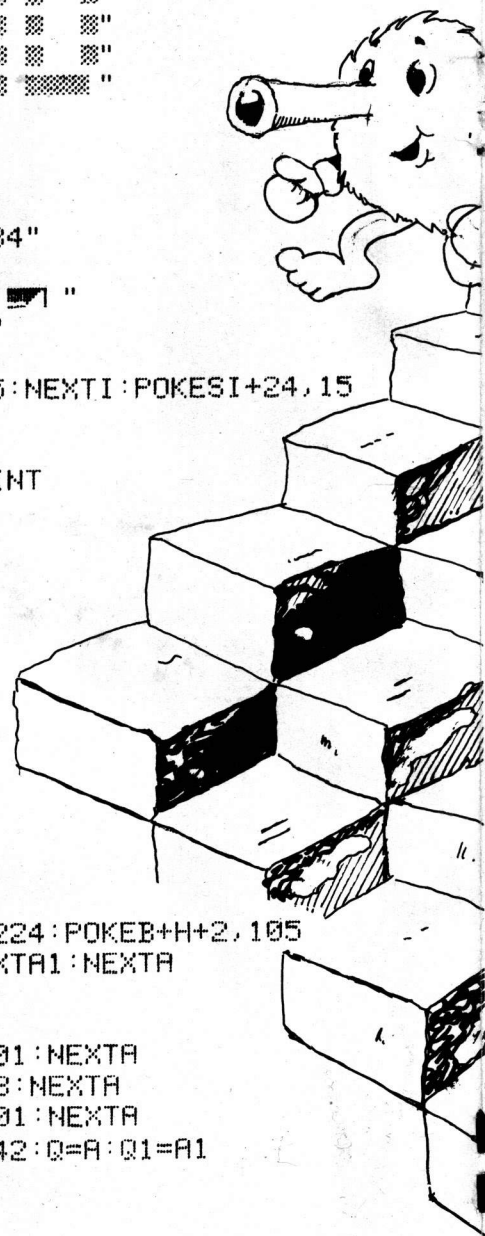
```

10 PRINT "Q":POKE53281,0:POKE53280,2
20 PRINT:PRINT" "
30 PRINT" "
40 PRINT" "
50 PRINT" "
60 PRINT" "
70 FORI=0TO1:PRINT" "
80 PRINT" "
90 PRINT" "
100 PRINT" "
110 PRINT" "
120 PRINT" "
130 PRINT" "
140 PRINT" "
150 PRINT" "
160 PRINT" "
170 PRINT" "
180 PRINT" (C) BY JESPER HANSEN 1984"
190 PRINT" "
200 PRINT" "
210 PRINT" "
220 GOSUB1130
230 SI=54272:FORI=5TO19STEP7:POKESI+I,15:POKESI+I+1,255:NEXTI:POKESI+24,15
240 POKE53280,2:PRINT"Q":FORI=1TO8:PRINT:NEXT
250 PRINT"VIL DU SPILLE MED":PRINT
260 PRINT"J=NYSTICK(PORT 1) ELLER J=ASTATUR":PRINT
270 B$=""
280 IFPEEK(203)=22THENPOKE198,0:GOTO320
290 IFPEEK(203)=34ORPEEK(203)=60THENB$="J":GOTO380
300 D=D+1:IFD>500THEND=0:GOTO10
310 GOTO280
320 PRINT"Q"
330 PRINT:PRINT"VILKEN TASTER SKAL VÆRE":PRINT
340 INPUT"-VENSTRE";L0$:PRINT
350 INPUT"-OP";R0$:PRINT
360 INPUT"-NED";L1$:PRINT
370 INPUT"-HØJRE";R1$
380 PRINT"Q":L=0:FA=3:P=0:RU=48
390 B=1160:F=55432:FORA=0TO6:GOTO1030
400 FORA1=0TO6:H=A*79+A1*42
410 FORI=0TO2:POKEF+H+I,1:NEXTI:POKEB+H,233:POKEB+H+1,224:POKEB+H+2,105
420 FORI=0TO1:POKEB+H+I+40,224:POKEF+H+I+40,2:NEXTI:NEXTA1:NEXTA
430 FORI=4TO18STEP7:POKESI+I,0:NEXTI:RESTORE
440 PRINT" SCORE:",P,"HI SCORE:",RE
450 B=1163:F=55435:FORA=0TO6:POKEF+A*42,1:POKEB+A*42,101:NEXTA
460 B=1454:F=55726:FORA=0TO6:POKEF+A*79,1:POKEB+A*79,78:NEXTA
470 B=1494:F=55766:FORA=0TO5:POKEF+A*79,1:POKEB+A*79,101:NEXTA
480 A=INT(RND(TI)*7):A1=INT(RND(TI)*7):S=1161+A*79+A1*42:Q=A:Q1=A1
490 POKES,222:POKES-40,170:IFA=0THENPOKES-40,101
500 SF=S-1024+55296
510 FORI=SF-1TOSF+1:POKEI,FA:NEXTI
520 POKE53280,FA
530 L=L+1:IFL>10THENL=10
540 FORI=1TOL
550 L0(I)=INT(RND(TI)*7):L1(I)=INT(RND(TI)*7):L(I)=1161+L0(I)*79+L1(I)*42
560 IFPEEK(L(I))<>224THEN550
570 POKEL(I),209:P0(I)=L0(I):P1(I)=L1(I):NEXTI
580 IFB$="J"THEN1050
590 GETA$:IFA$=""THEN640
600 IFA$=L0$THENA1=A1-1:GOTO800
610 IFA$=R1$THENA1=A1+1:GOTO800

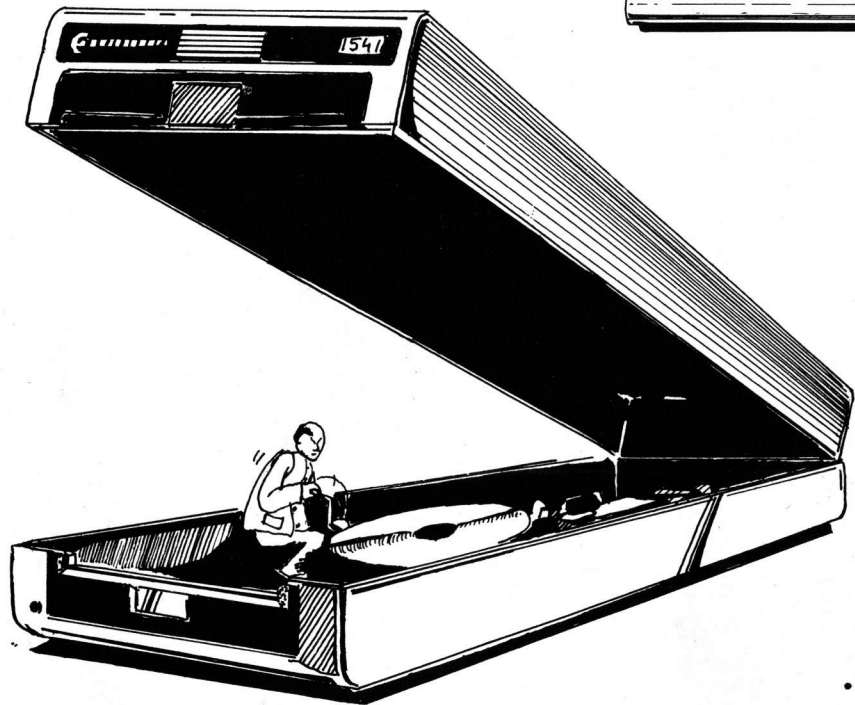
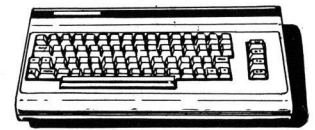
```

Vor ven Q-BERT skal styres rundt på de 49 trin, som Cheops pyramide består af. Hver gang han har været på en firkant, bliver feltet farvet. For at undgå Mumiens forbandelse, skal du få farvet alle felter. Et superspil til Commodore 64!!!

Jesper Hansen



Super-Bam



Med dette program har du nu chancen for endelig at få orden på dine disketter. Du kan ændre dit bamhoved, dit id. nr., samt ændre formatet på disketten. Endvidere kan du ændre din dos (disk operating system), så du writeprotecter din disk uden at påklæbe mærkat. Programmet er i øvrigt selvinstruerende og fortæller alt, hvad du vil vide i rem linjer.

Jan Toft

HCT

```

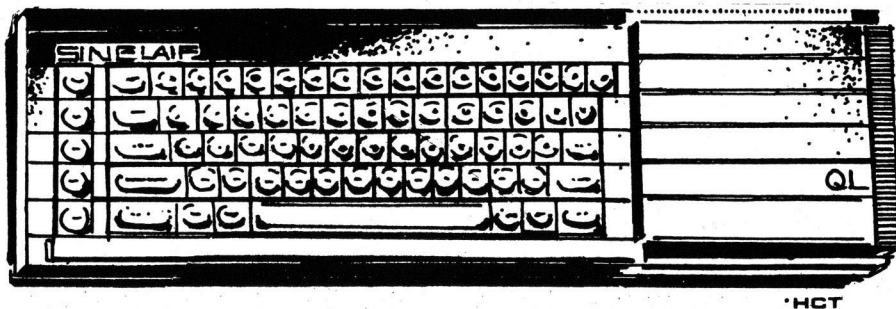
20 REM PROGRAM TIL AT ÆNDRE DISKETTE NAVN SAMT ID NR. SAMT DOS MÆRKE,
30 REM UDEN AT FORMÅTERE DISKETTEN OM
40 REM PÅ SKÆRMEN VISES TRE RÆKKER MED KLAMMER
50 REM HVER RÆKKE BESTÅR AF 4 KLAMMER
60 REM OVER FØRSTE KLAMME STAR DER DOS
70 REM HER KAN DU ÆNDRE DOS MÆRKET PÅ DISKETTEN
80 REM DET RIGTIGE MÆRKE ER 'A' ALLE ANDRE MÆRKER MEDFØRER !
90 REM AT DU IKKE KAN SKRIVE PÅ DISKETTEN
100 REM I KLAMME NR.2 SKRIVER DU DISKETTENS NAVN
110 REM I KLAMME NR.3 SKRIVER DU ID NR. BEMÆRK AT DETTE KUN ER OVERFLADISK
120 REM DA DET RIGTIGE ID NR STADIGVÆK VIL STA I HEADEREN PÅ HVER BLOK
130 REM DEN SIDSTE KLAMME HAR INGEN PRAKTISK BETYDNING
140 REM DU KAN SKRIVE LIGE HVAD DU LYSTER F.EKS DINE INITIALER
150 REM VED TRYK PÅ (F1) VISES GAMMEL BAMHOVED I DEN FØRSTE RÆKKE KLAMMER
160 REM (F1) KAN AKTIVERES PÅ ET VILKET SOM HELST TIDSPUNKT
170 REM DEN NÆSTE RÆKKE KLAMMER STAR DEN KUNSTIGE CURSER OG HER SKRIVER,
180 REM MÅN DET NYE BAMHOVED
190 REM I DEN TREDIE RÆKKE KLAMMER VISES DET NYE BAMHOVED
200 REM VARIABLER BRUGT I PROGRAM
210 REM L$ = ANTAL CURSER NED
220 REM X$ SAMT Y$ = BESTEMMER HVILKE TASTER DER MÅ TRYKES PÅ
230 REM C = TOMME FELTER I INPUT STRENG
240 REM Q = HVOR CURSEREN SKAL I INPUT
250 REM LN = MÅX KARAKTERER I INPUT
260 REM M = TÆLLER
270 REM P = BLOCK POINTER
280 REM A = VARIABEL I FOR OG NEXT LØKKE
290 REM C = ANTAL KARAKTERE I MID$
295 REM L = ANTAL CURSER NED
300 REM A$ = DEN ENDELIGE BAM
310 REM B$ = VARIABEL I INPUT
320 REM C2$+C3$+C4$ = TOMME FELTER BAMHOVEDET
330 REM F1$-F4$ = FEJL VARIABLER
340 REM F$ = HJÆLPE VARIABEL
350 REM E$ = LÆST BAMHOVED
360 REM A*(1) = DOS MÆRKE
370 REM A*(2) = DISKETTE NAVN
380 REM A*(3) = ID NR.
390 REM A*(4) = FORMAT
400 PRINT "":POKE2,3:REM SLETTER SKÆRM SAMT SÆTTER ANTAL AF KLAMMER
410 L$="":REM VARIABEL TIL CURSER POSITION
420 POKE53280,4:POKE53281,0:REM SKÆRM FÆRVER
430 PRINT " "
440 REM SKRIVER SKÆRM
450 REM
460 REM
470 PRINT "*****"
480 PRINT " N Y T B A M H O V E D "
490 PRINT "*****"
500 PRINT " ← R → S + S * CURSOR NED "
510 PRINTSPC(3) " DOS DISKETTE NAVN ID FORMAT "
520 FORX=1TOPEEK(2)
530 PRINT " "
540 PRINT " "
550 PRINT " "
560 PRINTLEFT(L$,19):PRINT " " TRYK F1 FOR SYN AF OPRINDELIG BAM "
570 PRINT " " TILVIRKET AF JAN TOFT D.1-10-1984 "
580 PRINT " " TIL OFFENTLIG BRUG "
590 REM
600 REM SKÆRM SLUT
610 REM
620 GOSUBS90

```

```

630 REM RUTINE TIL TJEK OG OPFYLDNING AF TOMME PLADSER I INPUT VARIABLERNE
640 C2$=""
650 C3$=""
660 C4$=C3$
670 C=LEN(C2$)-LEN(A*(2)):A*(2)=A*(2)+RIGHT(C2$,C)
680 C=LEN(C3$)-LEN(A*(3)):A*(3)=A*(3)+RIGHT(C3$,C)
690 C=LEN(C4$)-LEN(A*(4)):A*(4)=A*(4)+RIGHT(C4$,C)
700 A$=A*(2)+" "+A*(3)+" "+A*(4)+" "
710 REM
720 REM
730 P=144:GOSUB900:PRINT#2,A$:GOSUBB30:REM SKRIVER BAM
740 REM
750 P=2:GOSUB910:PRINT#2,A*(1):GOSUBB30:REM SKRIVER DOS MÆRKE
760 L=17:POKE646,8:GOSUB780:GOSUB1190:RUN410
770 REM LÆSER BAM OG HOPPER TIL PRINT RUTINE
780 P=144:GOSUB900:FORA=1TO34:GET#2,F$:E$=E$+F$:NEXT:POKE2,2:GOTO850
790 REM
800 REM
810 REM SKRIVER TIL DISKETTE OPERATIV SYSTEMET (DOS), AT DEN SKAL
820 IGNORERE DOS MÆRKET
830 PRINT#1,"M-W:CHR(1):CHR(1):CHR(65)
840 PRINT#1,"U2:":2:0:18:0:RETURN:REM SKRIVER PÅ SPOR 18 SECTOR 0
850 REM PRINTER BAM PÅ SKÆRMEN CURSOR T.H.
860 PRINTLEFT(L$,L)+" "+MID(E$,1,16):
870 PRINT " "+MID(E$,19,2):
880 PRINT " "+MID(E$,22,2):GOSUB1190:E$="":GOTO940
890 REM INITIALISERE OG ØBNER EN KANAL TIL DISKETTEN
900 GOSUB 1190:OPEN1,8,15,"":OPEN2,8,2,""
910 PRINT#1,"U1:":2:0:18:0:REM LÆSER SPOR 18 SECTOR 0
920 PRINT#1,"B-P:":2:P
930 RETURN
940 P=2:GOSUB900:GET#2,F$:PRINT " " +F$:RETURN
950 P=144:L=8:POKE646,5:GOSUB780:CLOSE1:PRINT " ":POKE211,CL:GOTO1010
960 X$=CHR(32):Y$=CHR(140):C=12:Q=3:LN=1:GOTO980
970 B$="":LN=14:Q=2
980 M=M+1:PRINT LEFT(L$,C):REM TÆLLER A*(M) EN OP
990 POKE211,Q:REM CURSER POSITION PÅ LINIEN
1000 REM KUNSTIG CURSER
1010 PRINT " " :FORA=1TO40:GETA$:IFA*(A)=""THEN1040
1020 NEXT:PRINT " " :FORA=1TO40:IFA*(A)=""THENNEXT:GOTO1010
1030 REM TEST FOR (F1)? HVIS JA! GEMMER CURSER POSITION OG LÆSER BAM IND
1040 IFASC(A$)=133THEN:CL=PEEK(211):GOTO950
1050 REM TEST FOR RETURN
1060 IFASC(A$)=13ANDQ<>13THENPRINT " " :GOTO1150
1070 IFASC(A$)>20THEN1100
1080 IFLEN(B$)=0THENNEXT:GOTO1010
1090 B$=LEFT(B$,LEN(B$)-1):PRINT " " :GOTO1010
1100 IFA*(X$ OR A$)Y$THENNEXT:GOTO1010 CURSOR T.H.
1110 IFLEN(B$)<LNTHENB$=B$+A$:PRINT " " +A$ " " :GOTO1010
1120 NEXT:GOTO1010
1130 REM POINTER TIL KUNSTIG CURSER
1140 REM
1150 IF Q=3 THENQ=9:LN=16:A*(M)=B$:B$="":GOTO980
1160 IF Q=9THENQ=29:LN=2:A*(M)=B$:B$="":GOTO980
1170 IF Q=29THENQ=35:LN=2:A*(M)=B$:B$="":GOTO980
1180 A*(M)=B$:IFA*(M)=35THENRETURN
1190 REM FEJL RUTINE
1200 CLOSE2:CLOSE1:OPEN1,8,15:INPUT#1,F1$,F2$,F3$,F4$
1210 IFF2$>"OK"THEN1230
1220 CLOSE2:CLOSE1:RETURN
1230 PRINT " " +LEFT(L$,12)+F1$+" "+F2$+" "+F3$+" "+F4$:CLOSE2:CLOSE1:END
READY.

```



QL Backup

Programmet foretager automatisk backup, en proces som ellers kræver en del tid og koncentration af brugeren. Vær under indtastningen opmærksom på linie 290, hvor der skal stå "tryk på ENTER.". A'et skrives ved at trykke CTRL SHIFT', desværre kan printeren ikke skrive tegnet som i QL er repræsenteret ved ASCII værdien 82H. Programmet er selvforklarende, så når det er tastet ind skrives blot RUN.□

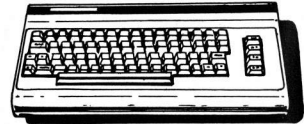
Michael Riber

```

y100 REMark          *** backup ***
110 REMark          af Michael Riber
120 :
130 CLS
140 PRINT "*** BACKUP *** Start ***"
150 PRINT "\"Kopierer alle filer fra mdv1 til mdv2."
160 PRINT "\"Placer de to kassetter og tryk ENTER.":PAUSE
170 PRINT "Forberedelse til backup."
180 :
190 OPEN_NEW #15,mdv1_dir_tmp
200 DIR #15,mdv1_
210 CLOSE #15
220 :
230 OPEN_IN #15,mdv1_dir_tmp
240 INPUT #15,catri9de_name$
250 INPUT #15,catri9de_status$
260 :
270 INPUT "\"Vil du formatere kassetten i mdv2 ? ";svar$
280 IF (svar$="J") OR (svar$="j") THEN
290     PRINT "Vent til mdv1 stopper, tryk s ENTER.":PAUSE
300     PRINT "Formatering : ";
310     FORMAT "mdv2_"& catri9de_name$
320 END IF
330 PRINT \
340 :
350 Antal_filer = 0
360 REPEAT kopier_fil
370 IF EOF (#15) THEN EXIT kopier_fil
380     INPUT #15,file_name$
390     IF file_name$ <> "dir_tmp" THEN
400         Antal_filer = Antal_filer + 1
410         PRINT "Kopierer ";file_name$
420         COPY "mdv1_"& file_name$ TO "mdv2_"& file_name$
430     END IF
440 END REPEAT kopier_fil
450 :
460 PRINT "\"Der blev kopieret "; Antal_filer; " filer."
470 CLOSE #15
480 DELETE mdv1_dir_tmp
490 :
500 PRINT "\"*** BACKUP *** Afsluttet ***"
```



Stroboskop



Et festligt program til Commodore 64, der sørger for lækre effekter under fester og lignende. Foruden stroboskoplyset er der 6 andre lyseffekter, som alle kan køre ud fra hovedmenuen.

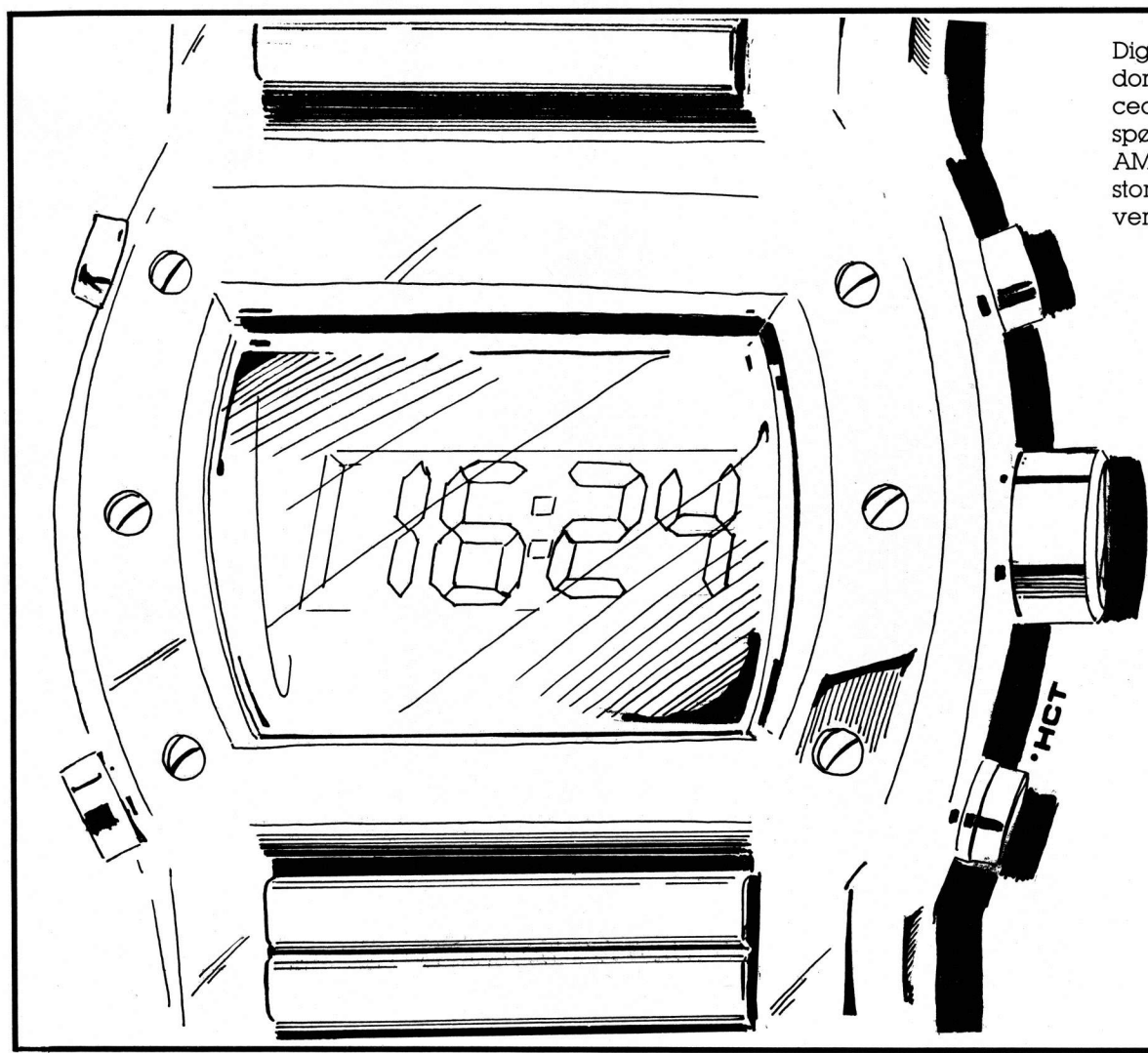
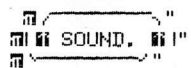
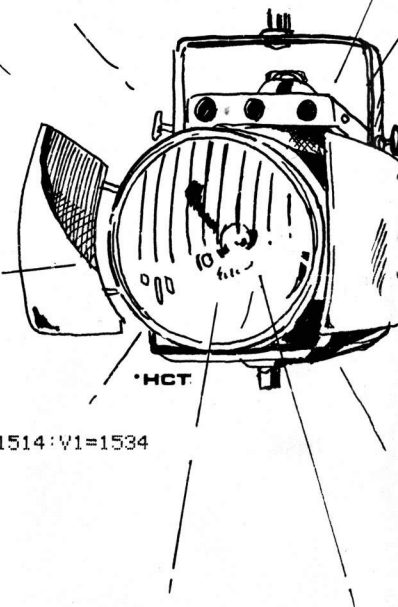
*Kurt Bennetsen og
Klaus Kortbæk*

```

0 PRINT "C": CLR
1 POKE 53281,0: POKE 53280,0: GOSUB 80
2 GET A$: IF A$="" THEN 2
3 IFA$=" " THEN 21
4 IFA$="." THEN 40
5 IFA$="|" THEN 50
6 IFA$=":" THEN 60
7 IFA$=";" THEN 100
8 IFA$="@" THEN 229
9 IFA$="↑" THEN PRINT "C": PRINT "m": GOTO 11
10 GOTO 2
11 PRINT CHR$(147)
12 PRINT "ER DU SIKKER (J/N)?"
13 GET SA$: IF SA$="" THEN 13
14 IF SA$="J" THEN PRINT "C": END
15 IF SA$="N" THEN RUN
16 GOTO 0
21 PRINT "C"
22 PRINT " "
23 PRINT " "
24 PRINT " "
    
```

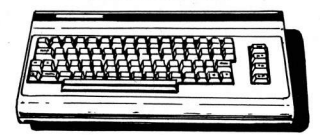
```

25 PRINT " "
26 PRINT " "
27 PRINT " "
28 PRINT " "
29 PRINT " "
30 PRINT " "
31 PRINT " "
32 GET B$: IF B$="" THEN 32
33 IF B$=" " THEN RUN
34 RUN
40 PRINT "C": REM ** STROBE LYS **
41 POKE 53281,0: POKE 53280,0
42 FOR A=1 TO 75: NEXT
43 POKE 53281,1
44 GET A1$: IF A1$="" THEN RUN
45 GOTO 41
50 REM ** FARVESTROBE **
51 PRINT "C": POKE 53281,0: POKE 53280,0
52 FOR A1=0 TO 15
53 POKE 53281,A1
54 FOR B=1 TO 50: NEXT B
55 POKE 53280,A1
56 FOR B=1 TO 50: NEXT B
57 GET A2$: IF A2$="" THEN RUN
58 NEXT A1
59 GOTO 51
60 REM ** COBALLS **
61 PRINT "C": POKE 53281,0: POKE 53280,0: V=1514: V1=1534
62 AB=INT(RND(1)*4)+1
63 IF AB=1 THEN V=V+40: V1=V1-40: F=2
64 IF AB=2 THEN V=V-40: V1=V1+40: F=5
65 IF AB=3 THEN V=V+1: V1=V1-1: F=6
66 IF AB=4 THEN V=V-1: V1=V1+1: F=7
67 IF V<0 THEN V=V+40
68 IF V>2023 THEN V=V-40
    
```



Digitaluret benytter Commodore 64's indbyggede interfacechip 6525. Programmet spørger om tiden, om det er AM/PM og viser herefter et stort display, der også angiver 1/10 sek.

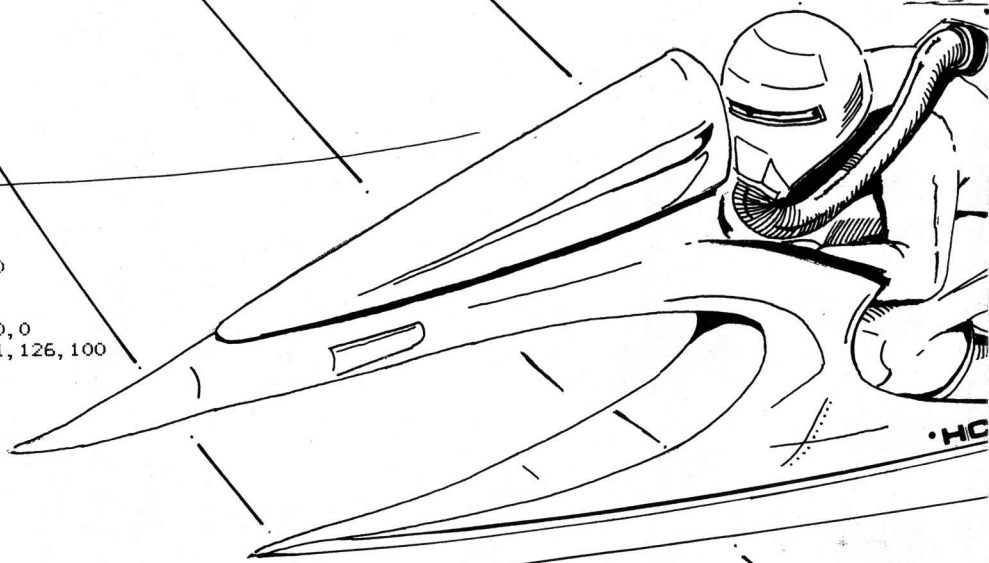
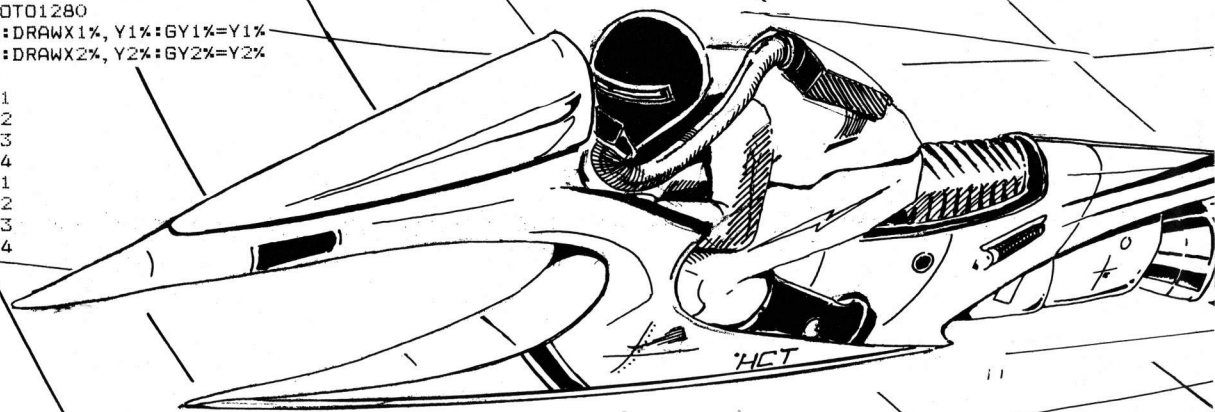
Henrik Larsen




```

10 REM      RUNDT PÅ GULVET for 2 spillere.
20 ON ERROR RUN
30 GOTO 370
40 IFPOINT (X1%, Y1%) (<)GOTO1130
50 IFPOINT (X2%, Y2%) (<)GOTO1280
60 GCOLO, 3:MOVEX1%, GY1%:DRAWX1%, Y1%:GY1%=Y1%
70 GCOLO, 6:MOVEX2%, GY2%:DRAWX2%, Y2%:GY2%=Y2%
80 IFW%=1GOTO180
90 IFADVAL (1)DIVD% (1B%=1
100 IFADVAL (1)DIVD% (1B%=2
110 IFADVAL (2)DIVD% (1B%=3
120 IFADVAL (2)DIVD% (1B%=4
130 IFADVAL (3)DIVD% (1A%=1
140 IFADVAL (3)DIVD% (1A%=2
150 IFADVAL (4)DIVD% (1A%=3
160 IFADVAL (4)DIVD% (1A%=4
170 GOTO260
180 IFINKEY (-104)B%=1
190 IFINKEY (-103)B%=2
200 IFINKEY (-105)B%=3
210 IFINKEY (-73)B%=4
220 IFINKEY (-83)A%=1
230 IFINKEY (-67)A%=2
240 IFINKEY (-98)A%=3
250 IFINKEY (-66)A%=4
260 SOUND1, -9, S1%, 1:S1%=S1%+1:SOUND2, -9, S2%, 1:S2%=S2%+1
270 ONA%GOTO280, 290, 300, 310
280 X1%=X1%+8:GOTO320
290 X1%=X1%-8:GOTO320
300 Y1%=Y1%-8:GOTO320
310 Y1%=Y1%+8
320 ONB%GOTO330, 340, 350, 360
330 X2%=X2%+8:GOTO40
340 X2%=X2%-8:GOTO40
350 Y2%=Y2%-8:GOTO40
360 Y2%=Y2%+8:GOTO40
370 MODE 1:VDU 23, 1, 0; 0; 0; 0;
380 VDU 23, 66, 126, 51, 51, 62, 51, 51, 126, 0
390 VDU 23, 68, 126, 51, 51, 51, 51, 51, 126, 0
400 VDU 23, 86, 99, 99, 99, 54, 54, 28, 28, 0
410 VDU 23, 91, 31, 60, 108, 127, 108, 108, 111, 0
420 VDU 23, 92, 62, 103, 111, 123, 115, 99, 62, 0
430 VDU 23, 93, 28, 0, 62, 99, 127, 99, 99, 0
440 ENVELOPE 2, 0, 0, 0, -1, 1, 1, 255, 0, 0, 0, 0, 0
450 ENVELOPE 3, 3, 0, 0, 0, 0, 0, 126, -2, -1, -1, 126, 100
460 VDU 19, 3, 6, 0, 0, 0
470 CLS:GCOL 0, 2
480 DRAW 1279, 0:DRAW 1279, 1023
490 DRAW 0, 1023:DRAW 0, 0
500 MOVE 4, 0:DRAW 4, 1023
510 MOVE 1275, 0:DRAW 1275, 1023
520 MOVE 12, 8:DRAW 1266, 8:DRAW 1266, 1012
530 DRAW 12, 1012:DRAW 12, 8
540 MOVE 16, 8:DRAW 16, 1012
550 MOVE 1262, 8:DRAW 1262, 1012
560 COLOUR 1
570 PRINTTAB (10, 3) "*****"
580 PRINTTAB (10, 4) "*"
590 PRINTTAB (10, 5) "*"
600 PRINTTAB (10, 6) "*"
610 PRINTTAB (10, 7) "*****"
620 COLOUR 2:PRINTTAB (19, 4) "."
630 PRINTTAB (12, 5) "RUNDT PÅ GULVET"
640 COLOUR 3
650 PRINTTAB (12, 10) "For 2 spillere."
660 COLOUR 2:S=174
670 FOR X=1 TO 38 STEP 2
680     SOUND 1, -11, S, 1
690     SOUND 2, -11, S-1, 1
700     S=S-4
710     PRINTTAB (X, 12) "___";
720     NEXT:COLOUR 2
730 PRINTTAB (6, 15) "GUL styrer med A Z X C"
740 COLOUR 3
750 PRINTTAB (8, 16) "."
760 PRINTTAB (6, 17) "BLA styrer med , . / : "
770 COLOUR 1
780 PRINTTAB (3, 21) "DET GÅLDER OM IKKE AT RAMME NOGET."
790 PRINT:COLOUR 2
800 FOR X=1 TO 38
810     PRINTTAB (X, 24) "___";
820     NEXT:COLOUR 3
830 PRINTTAB (8, 28) "Bruges JOYSTICK ? (J/N)"
840 FOR S=-15 TO 0 STEP 0.3
850     SOUND 1, S, 80, 1
860     SOUND 2, S, 81, 1
870     NEXT
880 Q$=GET$
890 IF Q$="J" W%=0:GOTO 920
900 IF Q$="N" W%=1:GOTO 920
910 GOTO 880
920 MODE 2
930 VDU 23, 1, 0; 0; 0; 0;
940 AP=0:BP=0:OM=10:D%=20000
950 COLOUR 144:CLS:GCOL 0, 1:MOVE 0, 4

```

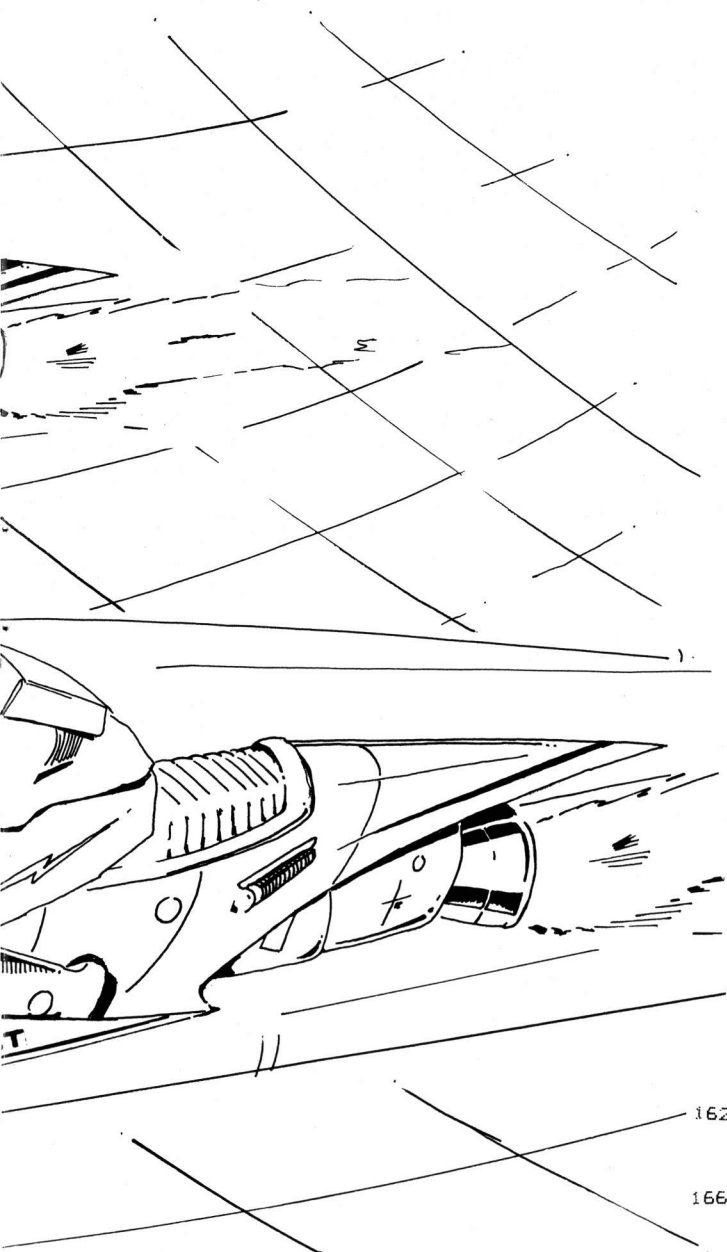


Rundt på

»Rundt på gulvet« er et TRON-lignende spil beregnet til en 32K BBC computer. Programmet kører kun med to spillere, og du kan selv vælge, om det skal foregå med joystick eller fra tastaturet. I øvrigt er programmet selvforklarende.

Hans Hansen





gulvet

```

960 DRAW 0,1023:DRAW 1279,1023
970 DRAW 1279,4:DRAW 0,4
980 X1%=300:Y1%=500:GY1%=Y1%
990 X2%=979:Y2%=500:GY2%=Y2%
1000 AX=1:BX=2:S1%=0:S2%=100
1010 FOR Q=1 TO 2000:NEXT
1020 FOR I=3 TO 1 STEP -1
1030 PRINTTAB(9,16);I
1040 FOR S=-15 TO 0
1050 SOUND 1,S,60,2
1060 SOUND 2,S,61,2
1070 NEXT
1080 FOR J=1 TO 1400:NEXT
1090 NEXT
1100 PRINTTAB(9,16) " "

```

```

1110 *FX 15,0
1120 GOTO 270
1130 IF POINT(X2%,Y2%) < 0 THEN 1410
1140 SOUND 17,0,200,1
1150 SOUND 16,3,7,1
1160 VDU 19,0,3,0,0,0
1170 FOR K=1 TO 100:NEXT
1180 VDU 19,0,0,0,0,0
1190 GCOL 0,3
1200 FOR I=0 TO 8
1210 RX%=RND(100)-1:RY%=RND(100)-1
1220 MOVE X1%,Y1%:PLOT 1,RX%,RY%
1230 MOVE X1%,Y1%:PLOT 1,-RX%,RY%
1240 MOVE X1%,Y1%:PLOT 1,RX%,-RY%
1250 MOVE X1%,Y1%:PLOT 1,-RX%,-RY%
1260 NEXT
1270 AP=AP+1:GOTO 1600
1280 SOUND 17,0,200,1
1290 SOUND 16,3,7,1
1300 VDU 19,0,6,0,0,0
1310 FOR K=1 TO 100:NEXT
1320 VDU 19,0,0,0,0,0
1330 FOR IX=0 TO 8
1340 RX%=RND(100)-1:RY%=RND(100)-1
1350 MOVE X2%,Y2%:PLOT 1,RX%,RY%
1360 MOVE X2%,Y2%:PLOT 1,-RX%,RY%
1370 MOVE X2%,Y2%:PLOT 1,RX%,-RY%
1380 MOVE X2%,Y2%:PLOT 1,-RX%,-RY%
1390 NEXT
1400 BP=BP+1:GOTO 1600
1410 SOUND 17,0,200,1
1420 SOUND 16,3,7,1
1430 FOR L=1 TO 2
1440 VDU 19,0,7,0;0
1450 FOR K=1 TO 88:NEXT
1460 VDU 19,0,0,0;0
1470 FOR M=1 TO 88:NEXT
1480 NEXT
1490 FOR IX=0 TO 8
1500 RX%=RND(100)-1:RY%=RND(100)-1
1510 GCOL 0,3:MOVE X1%,Y1%:PLOT 1,RX%,RY%
1520 GCOL 0,6:MOVE X2%,Y2%:PLOT 1,RX%,RY%
1530 GCOL 0,3:MOVE X1%,Y1%:PLOT 1,-RX%,RY%
1540 GCOL 0,6:MOVE X2%,Y2%:PLOT 1,-RX%,RY%
1550 GCOL 0,3:MOVE X1%,Y1%:PLOT 1,RX%,-RY%
1560 GCOL 0,6:MOVE X2%,Y2%:PLOT 1,RX%,-RY%
1570 GCOL 0,3:MOVE X1%,Y1%:PLOT 1,-RX%,-RY%
1580 GCOL 0,6:MOVE X2%,Y2%:PLOT 1,-RX%,-RY%
1590 NEXT:FOR J=1 TO 4444:NEXT
1600 COLOUR 150:COLOUR 0
1610 PRINTTAB(17,0);" "
1620 IF AP=0 THEN PRINTTAB(17,1);" 0 " ELSE PRINTTAB(17,1);" ";AP;" "
1630 PRINTTAB(17,2);" "
1640 COLOUR 147
1650 PRINTTAB(0,0);" "
1660 IF BP=0 THEN PRINTTAB(0,1);" 0 " ELSE PRINTTAB(0,1);" ";BP;" "
1670 PRINTTAB(0,2);" "
1680 OM=OM-1
1690 IF OM=1 THEN 1750
1700 IF OM=0 THEN 1810
1710 COLOUR 148:PRINTTAB(2,29);" "
1720 COLOUR 7:PRINTTAB(2,30);" ";OM;" omg. tilbage "
1730 PRINTTAB(2,31);" ";
1740 GOTO 1790
1750 COLOUR 145:COLOUR 7
1760 PRINTTAB(2,29);" "
1770 PRINTTAB(2,30);" Sidste omgang! "
1780 PRINTTAB(2,31);" ";
1790 FOR Q=1 TO 7000:NEXT
1800 CLS:GOTO 950
1810 IF AP=BP COLOUR 144:PRINTTAB(5,2) " "
1820 IF AP=BP COLOUR 2:PRINTTAB(5,3) " UAFGJORT "
1830 IF AP=BP COLOUR 144:PRINTTAB(5,4) " "
1840 IF AP<BP COLOUR 144:PRINTTAB(4,2) " "
1850 IF AP<BP COLOUR 3:PRINTTAB(4,3) " GUL VINDER "
1860 IF AP<BP COLOUR 3:PRINTTAB(4,4) " "
1870 IF AP>BP COLOUR 144:COLOUR 6:PRINTTAB(4,2) " ."
1880 IF AP>BP PRINTTAB(4,3) " BLA VINDER "
1890 IF AP>BP PRINTTAB(4,4) " "
1900 COLOUR 145:FOR I=15 TO 17
1910 PRINTTAB(7,I) " "
1920 NEXT
1930 COLOUR 7
1940 PRINTTAB(7,16) " SLUT "
1950 IF WX=0 GOTO 2000
1960 COLOUR 148:PRINTTAB(0,28) " "
1970 COLOUR 7:PRINTTAB(0,29) "Tryk A og : samtidig"
1980 PRINTTAB(0,30) " ";
1990 GOTO 2050
2000 COLOUR 148:PRINTTAB(0,28) " "
2010 COLOUR 7:PRINTTAB(0,29) " Tryk begge KNAPPER "
2020 PRINTTAB(0,30) " ";
2030 IF (ADVAL(0)AND3)=3 GOTO 920
2040 GOTO 2030
2050 IF INKEY=66 AND INKEY=73 GOTO 920
2060 GOTO 2050

```

Rendez-Vous er et rumspil til Texas 99 4A, og er på højde med de spil, der sælges i forretningerne.

Lars Østerballe



Rende

```

100 !!!!!!!!!!!!!!!!!!!!!
/ RENDEZ-VOUS 2.6 /
/ VERSION 1.1 /
/ LARS ØSTERBALLE 19 /
B4 / !!!!!!!!!!!!!!!!!!!!!
!!!!
110 CALL CLEAR :: CALL SCREE
N(2)
120 C=.1 :: BR=400 :: RANDOM
IZE :: CALL PEEK(-28672,TALE
)! ER TALESYNTHESIZEREN
TILSLUTTET?
130 DATA FIVE,FOUR,THREE,TWO
ONE,ZERO,Q
140 CALL COLOR(1,16,1,2,7,1,
3,7,1,4,7,1):: FOR F=5 TO 12
:: CALL COLOR(F,13,1)::
NEXT F
150 CALL COLOR(0,11,11):: GO
SUB 820 :: GOSUB 1340 :: CAL
L MAGNIFY(3):: GOSUB 175
0
160 !//SKEARM-GRAFIK+FASE//
170 CALL CLEAR :: FOR F=1 TO
15 :: CALL HCHAR(RND*23+1,R
ND*31+1,33):: NEXT F ::
CALL COLOR(0,11,11):: CALL H
CHAR(24,1,30,32)
180 V=0 :: A=0 :: B=0 :: CAL
L SPRITE(#1,100,15,1,128,#2,
96,7+FL*4,185,RND*255+1,
0,(RND-RND)*10)
190 CALL SOUND(-10,110,0)
200 IF FL=0 THEN 240
210 FLA=1 :: IF C>.6 THEN C=
.1
220 CALL COLOR(0,8-FL,8-FL)
230 FOR F=3 TO 3+FL :: CALL
SPRITE(#F,104,RND*13+3,F*45-
75,120,0,(RND-RND)*(C*10
+20)):: NEXT F
240 DISPLAY AT(1,1):USING "S
CORE #### TOP ####"
:POINT,HIGH
250 DISPLAY AT(3,1):USING "F
ASE NR. #. #":FL+1+C;
260 !!!!!/LANDINGSRUTINE////
270 CALL KEY(3,K,S):: A=A-(K
=NED)+(K=OP)+C :: B=B-(K=HOJ
)+(K=VEN)
280 DISPLAY AT(3,19):USING "
BRINT ####":BR
290 IF BR<=0 THEN 650
300 IF K=OP THEN CALL SOUND(
-350,-5,20):: BR=BR-5 ELSE I
F K=VEN OR K=HOJ THEN CA
LL SOUND(-350,-6,23):: BR=BR
-1
310 CALL POSITION(#1,X,Y)::
IF X>169 AND X<200 THEN CALL
MOTION(#1,0,0,#2,0,0)::
CALL COINC(ALL,V)
320 IF X>240 THEN GOSUB 620
330 IF V THEN 490 ELSE IF X>
169 THEN M$=" KOLLISION
MED JORD" :: GOTO 370
340 IF X<169 THEN CALL COINC
(ALL,V):: IF V THEN M$=" KO
LLISION MED ASTEROIDE" :
: GOTO 370
350 CALL MOTION(#1,A,B):: GO
TO 270
360 !!!!!/SPILLET ER SLUT////
370 CALL DELSPRITE(#2,#3,#4,
#5,#6):: GOSUB 690 :: FLAG=0
:: KRASH=0
380 IF TALE THEN CALL SAY("S
ORRY+PARTNER")
390 HIGH=MAX(HIGH,POINT)
400 DISPLAY AT(9,1):M$;
410 IF POINT=HIGH AND POINT<
>0 THEN GOSUB 1460 ELSE IF B
R>0 THEN CALL PAUSE(500)

```

```

420 DISPLAY AT(9,1)
430 POINT=0 :: BR=400 :: C=.
1 :: FL=0 :: FLA=0 :: IF TAL
E THEN CALL SAY("PRESS+A
1+KEY+TO+PLAY+AGAIN")
440 DISPLAY AT(11,4):"TRYK R
EDD ELLER BACK";
450 DISPLAY AT(9,10):"GAME O
VER" :: FOR F=1 TO 15 :: CAL
L KEY(3,K,S):: IF K=6 TH
EN 170 ELSE IF K=15 THEN GOS
UB 1680 :: GOTO 170
460 NEXT F :: DISPLAY AT(9,1
):: FOR F=1 TO 15 :: CALL KE
Y(3,K,S):: IF K=6 THEN 1
70 ELSE IF K=15 THEN GOSUB 1
680 :: GOTO 170
470 NEXT F :: GOTO 450
480 !/ER LANDING FOR HAARD?/
490 IF A>=4 THEN M$="
FART FOR HØJ" :: GOTO 370
500 CALL POSITION(#1,X,Y,#2,
XX,YY):: CALL LOCATE(#1,169,
Y):: BO=ABS(16-ABS(YY-Y)
)
510 DISPLAY AT(9,9):USING "B
ONUS ####:BO:: FOR F=0 TO 3
0 :: CALL SOUND(-99,110,
F,600,F,1000,F):: NEXT F ::
DISPLAY AT(9,1)
520 POINT=POINT+BO :: DISPLA
Y AT(1,1):USING "SCORE ##
## TOP ####":POINT
,HIGH
530 FOR F=110 TO 110+BO*15 S
TEP 4 :: DISPLAY AT(3,25):US
ING "####":F+BR-110 :: C
ALL SOUND(-199,F*2,10):: NEX
T F :: BR=BR+F-114
540 IF FLAG THEN 570
550 GOSUB 1020
560 IF BR<=0 THEN 650 ELSE I
F KRASH THEN 370 ELSE IF SUC
CES THEN 510
570 IF C<.7 THEN C=C+.1
580 IF C>.6 AND FL<2 THEN FL
=FL+1
590 IF TALE THEN CALL SAY("G
OOD+WORK")
600 FLAG=0 :: SUCCES=0 :: CA
LL PAUSE(300):: CALL DELSPRI
TE(ALL):: GOTO 180
610 !!!!!/FOR HOEJT!!!!
620 DISPLAY AT(9,9):"HIMMELS
TREJF";
630 A=-A :: CALL MOTION(#1,A
,B):: FOR F=0 TO 30 STEP 2 :
: CALL SOUND(-99,523,F):
: NEXT F :: DISPLAY AT(9,9):
:
640 CALL POSITION(#1,X,Y)::
RETURN
650 DISPLAY AT(3,19):USING "
BRINT ####":0
660 GOSUB 740 :: DISPLAY AT(
9,1)
670 GOSUB 680 :: FLAG=0 :: K
RASH=0 :: GOTO 390
680 !!!!!/EKSPLOSION!!!!
690 CALL POSITION(#1,X,Y)::
CALL DELSPRITE(#5,#6):: IF Y
>248 THEN Y=248
700 CALL MAGNIFY(1):: CALL S
PRITE(#1,100,7,X,Y,-20,-20,#
2,101,5,X+8,Y,-5,-40)
710 CALL SPRITE(#3,102,3,X,Y
+8,-20,20,#4,103,11,X+8,Y+8,
-5,40)
720 FOR F=0 TO 30 :: CALL SO
UND(-99,-5,F):: NEXT F :: CA
LL DELSPRITE(ALL):: CALL
MAGNIFY(3):: RETURN

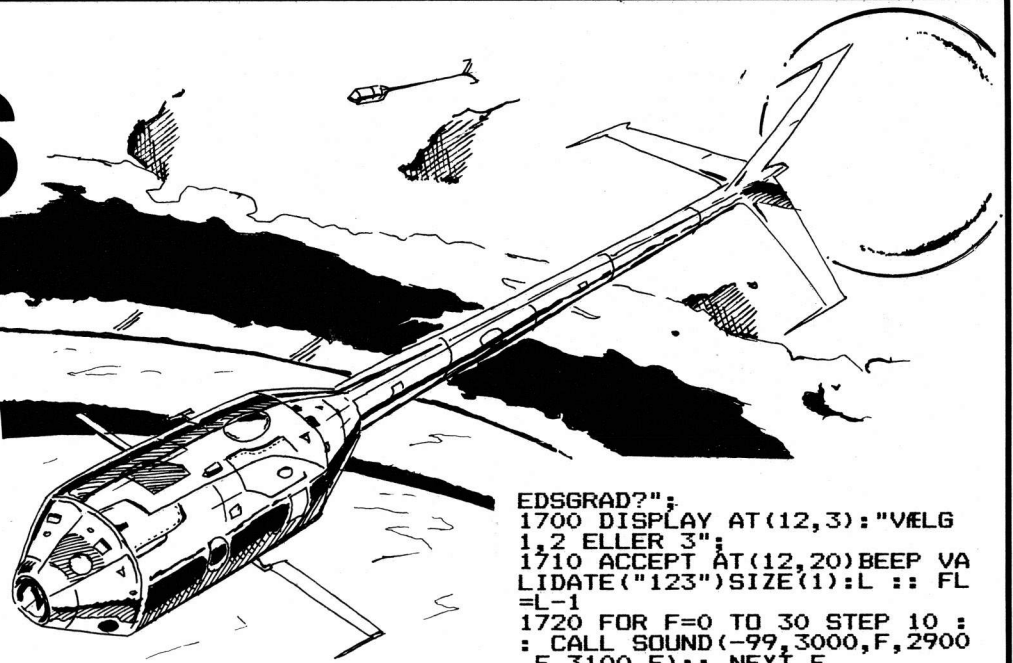
```

```

730 !!!!!/INTET BRAENDSTOF///
740 DISPLAY AT(9,9):"NUL BRÆ
NDSTOF"
750 FOR F=0 TO 30 :: CALL SO
UND(-99,262,F):: NEXT F :: M
$=""
760 CALL DELSPRITE(#3,#4,#5)
770 FOR F=1 TO 90
780 CALL MOTION(#1,A+F,B)
790 CALL POSITION(#1,X,Y)::
IF X>169 AND X<200 THEN RETU
RN
800 CALL POSITION(#1,X,Y)::
IF X>169 AND X<200 THEN RETU
RN
810 NEXT F :: RETURN
820 !!!!!/KARAKTERER!!!!
830 CALL CHAR(33,"1")
840 CALL CHAR(40,"001C387070
70381C00381C0E0E0E1C38",63,"
007CEEEEE0E3C003C")
850 CALL CHAR(100,"804023173
F6C6C1C6FEFEFFFE34747E0090AC
4FBF8949498F6F7F7FFC7E2E
207")
860 CALL CHAR(104,"0000136CB
3C0B8C6818040483403010000C03
00E916112C5398545522246B
8")
870 CALL CHAR(64,"3C42BDA1A1
BD423C00FEFEFEFEFEFEFEFEFE
EEFCFEFEFE007CEFEFEFEFEFE
7C")
880 CALL CHAR(68,"00FCFEFEFE
EEFEFCFEFEFEFEFEFEFEFEFE
E0FE0E0E0E00FEFEFEFEFEFE
FE")
890 CALL CHAR(72,"00EEFEFEFE
EEFEFEFEFEFE38383838FE00FE
0E0E0E0E0E0E0E0E0E0E0E0E
EE")
900 CALL CHAR(76,"00E0E0E0E0
EEFE0E0E0E0E0E0E0E0E0E0E
FEFEFEFEFE6007CEFEFEFEFE
7C")
910 CALL CHAR(80,"00FEFEFEFE
FE0E00FEFEFEFEFEFEFEFEFE
EEFEFCFEFEFE00FEFEFEFEFE
FE")
920 CALL CHAR(84,"00FEFEBA38
38383800EEFEFEFEFEFEFEFEFE
EEFEFE6C3800EEFEFEFEFEFE
FE")
930 CALL CHAR(88,"00EEFEFE38
FEFEFE0E0E0E0E0E0E0E0E0E
3E7CF8FEFE")
940 CALL CHAR(48,"007CEFEFE
EEEE7C003878F8383838FE00FE
0E0E0E0E0E0E0E0E0E0E0E0E
FE")
950 CALL CHAR(52,"00EEFEFEFE
FE0E0E0E0E0E0E0E0E0E0E0E
E0FEFEFEFE00FEFEFE0E1C3838
38")
960 Q$="FFFFFFFFFFFFFFFF" ::
B$=RPT$("0",16):: CALL CHAR
(96,Q$&B$&Q$&B$)
970 CALL CHAR(56,"00FEFEFEFE
EEFEFE00FEFEFEFEFEFEFEFE")
980 CALL CHAR(108,"070402313
F2ABF7F7FBF2131190D0703E0204
0BCFC54DFEFDFCF8F8F0E
OC")
990 CALL CHAR(91,"007EDAD8FC
DBDADE",92,"007AE4EEFEFE4EBC
",93,"0038007CEFEFEFEFE")
1000 RETURN
1010 !!!!!/SAMMENKOBLING!!!!
1020 FLAG=1 :: A=0 :: B=0 ::
CALL SPRITE(#6,108,RND*13+3
,26,RND*255+1,0,(RND-RND
)*(C*10+13))

```

ez-Vous



```
1030 DISPLAY AT(9,9):"RENDEZ
-VOUS" :: FOR F=0 TO 30 :: C
ALL SOUND(-99,1200,F)::
NEXT F
1040 RESTORE :: GOSUB 1240
1050 CALL KEY(3,K,S):: IF K<
>OP THEN 1050 ELSE DISPLAY A
T(9,1)
1060 CALL KEY(3,K,S):: A=A-(
K=NED)+(K=OP)+C :: B=B-(K=HO
J)+(K=VEN)
1070 IF K=OP THEN CALL SOUND
(-350,-5,20):: BR=BR-5 ELSE
IF K=VEN OR K=HOJ THEN C
ALL SOUND(-350,-6,23):: BR=B
R-1
1080 DISPLAY AT(3,19):USING
"BRINT ####":BR
1090 IF BR<=0 THEN RETURN
1100 CALL POSITION(#1,X,Y)
1110 IF X>169 AND X<200 THEN
KRASH=1 :: M$=" KOLLISI
ON MED JORD" :: RETURN
1120 CALL POSITION(#1,X,Y)
1130 IF X<47 THEN 1180
1140 CALL COINC(ALL,V):: IF
V AND X<169 THEN KRASH=1 ::
M$=" KOLLISION MED ASTE
ROIDE" :: RETURN
1150 IF X<47 THEN 1180
1160 IF X>240 THEN GOSUB 620
1170 CALL MOTION(#1,A,B):: G
OTO 1060
1180 CALL COINC(ALL,V):: IF
NOT V THEN 1160
1190 CALL MOTION(#1,0,0,#6,0
,0):: CALL POSITION(#1,X,Y,#
6,XX,YY)
1200 IF A<-4 OR A>4 THEN KRA
SH=1 :: M$=" FART FOR
HØJ" :: RETURN
1210 IF X<26 THEN KRASH=1 ::
M$=" KOLLISION MED SKIB
" :: RETURN
1220 SUCCES=1 :: BO=ABS(16-A
BS(YY-Y)):: RETURN
1230 !!!!!/NEDTÆLLING!!!!!!
1240 FOR F=5 TO 0 STEP -1
1250 DISPLAY AT(9,9):USING "
TÆLLING #":F;
1260 IF TALE THEN READ A$ ::
CALL SAY(A$)ELSE GOSUB 1320
1270 NEXT F
1280 DISPLAY AT(9,9):" LIFT
OFF "
1290 IF TALE THEN CALL SAY("
CONNECTED")ELSE GOSUB 1320
1300 RETURN
1310 !!!!!/BIP!!!!!!
1320 CALL SOUND(-10,2000,0):
: FOR P=1 TO 200 :: NEXT P :
: RETURN
1330 !!!!!/START-BILLEDE!!!!
1340 CALL HCHAR(1,3,30,30)::
CALL HCHAR(24,3,30,30):: CA
LL VCHAR(2,3,30,22):: CA
LL VCHAR(2,32,30,22)
1350 DISPLAY AT(7,8):"RENDEZ
-VOUS 2.6";
1360 DISPLAY AT(11,13):"@ 19
84":: DISPLAY AT(13,8):"LAR
S ØSTERBALLE";
1370 IF TALE THEN CALL SAY("
WELL+COME+TO+THE1+SPACE+COMM
AND")
1380 DISPLAY AT(17,7):"VÆLG
TASTER (J/N)?"
1390 CALL KEY(3,K,S):: IF K<
>74 AND K<>78 THEN 1390
1400 IF K=74 THEN GOSUB 1550
:: GOTO 1420
```

```
1410 OP=69 :: NED=88 :: HOJ=
68 :: VEN=83 :: DISPLAY AT(1
7,7):" BRUG PILENE.
":: CALL PAUSE(200):: DISPL
AY AT(17,7):"
"
1420 DISPLAY AT(17,3):"
VÆLG NIVEAU? "
1430 CALL KEY(3,K,S):: IF S<
=0 THEN 1430 ELSE IF K=74 TH
EN GOSUB 1680
1440 CALL CLEAR :: RETURN
1450 !//HIGHSCORE-MELDING//
1460 FOR F=1 TO 3
1470 DISPLAY AT(1,1):USING "
SCORE ####":POINT;
1480 FOR G=0 TO 30 STEP 2
1490 CALL SOUND(-99,330,6)
1500 NEXT G
1510 DISPLAY AT(1,1):RPT$("
" 13);
1520 CALL PAUSE(100):: NEXT
F
1530 DISPLAY AT(1,1):USING "
SCORE #### TOP ####
":POINT,HIGH:: RETURN
1540 !!!!!/VÆLG TASTER!!!!
1550 CALL CLEAR
1560 CALL HCHAR(1,3,30,30)::
CALL HCHAR(24,3,30,30):: CA
LL VCHAR(2,3,30,22):: CA
LL VCHAR(2,32,30,22)
1570 DISPLAY AT(3,3):"HVLKE
N TAST FOR:"
1580 DISPLAY AT(5,3):"OP";
1590 ACCEPT AT(5,11)VALIDATE
(UALPHA,NUMERIC,";/=")BEEP S
IZE(1):OP$ :: OP=ASC(OP$
)
1600 DISPLAY AT(7,3):"NED";
1610 ACCEPT AT(7,11)VALIDATE
(UALPHA,NUMERIC,";/=")BEEP S
IZE(1):NED$ :: NED=ASC(N
ED$)
1620 DISPLAY AT(9,3):"VENSTR
E";
1630 ACCEPT AT(9,11)VALIDATE
(UALPHA,NUMERIC,";/=")BEEP S
IZE(1):VEN$ :: VEN=ASC(V
EN$)
1640 DISPLAY AT(11,3):"HØJRE
";
1650 ACCEPT AT(11,11)VALIDAT
E(UALPHA,NUMERIC,";/=")BEEP
SIZE(1):HOJ$ :: HOJ=ASC(
HOJ$)
1660 GOSUB 1320 :: RETURN
1670 !//VÆLG SVAERHEDSGRAD//
1680 CALL CLEAR :: CALL HCHA
R(1,3,30,30):: CALL HCHAR(24
,3,30,30):: CALL VCHAR(2
,3,30,22):: CALL VCHAR(2,32
,30,22)
1690 DISPLAY AT(10,3):"SVÆRH
```

*HCT

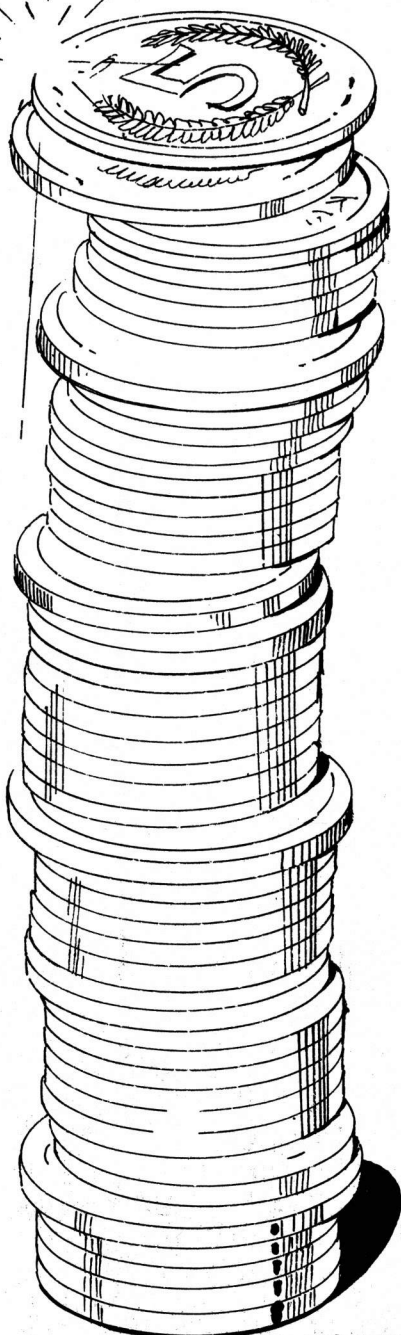
```
EDSGRAD?";
1700 DISPLAY AT(12,3):"VÆLG
1,2 ELLER 3";
1710 ACCEPT AT(12,20)BEEP VA
LIDATE("123")SIZE(1):L :: FL
=L-1
1720 FOR F=0 TO 30 STEP 10 :
: CALL SOUND(-99,3000,F,2900
,F,3100,F):: NEXT F
1730 RETURN
1740 !!!!!/INSTRUKTIONER!!!!
1750 CALL SCREEN(15):: DISPL
AY AT(1,1):" RENDEZ-VOU
S 2.6
~~~~~
1760 DISPLAY AT(4,1):"LAND D
IT RUMSKIB SA SIKKERT
SO
M MULIGT PA KLODEN 'QUAQ'."
1770 DISPLAY AT(8,1):"FOR AT
FA FLEST POINTS, SKAL
DU
LANDE NØJAGTIGT PA DEN"
1780 DISPLAY AT(12,1):"LILLE
PLATFORM, DER BEVÆGER
S
IG PA OVERFLADEN."
1790 DISPLAY AT(16,1):"LAND
IKKE FOR HURTIGT OG
I
SÆR IKKE VED SIDEN AF..."
1800 DISPLAY AT(20,1):"NÅR D
U ER LANDET ER DIN
N
ÆSTE OPGAVE AT LAVE EN GOD"
1810 DISPLAY AT(24,1):"SAMME
NKOBLING MED 'EAGLEN'"
1820 GOSUB 1910
1830 DISPLAY AT(1,1):"HER SK
AL DU LIGELEDES VÆRE
FO
RSIGTIG, OG LAND IKKE PA"
1840 DISPLAY AT(5,1):"TOPPEN
AF FARTØJET....
FO
R HVER AF DISSE FASER DU"
1850 DISPLAY AT(9,1):"FULDFØ
RER, VIL TYNGEKRAFTEN
ST
IGE(SE 'FASE NR'), OG NÅR"
1860 DISPLAY AT(13,1):"DU HA
R GENNEMFØRT SEKS FASER
B
LIVER DU SENDT VIDERE TIL"
1870 DISPLAY AT(17,1):"NÆSTE
SKÆRM, HVOR NYE FOR-
H
INDRINGER DUKKER OP."
1880 DISPLAY AT(21,1):"
HELD OG LYKKE..."
1890 GOSUB 1910 :: CALL SCRE
EN(2) :: RETURN
1900 !!!!!/TILFÆLDIGE LYDE//
1910 TQ=RND*1000+140 :: FOR
F=0 TO 30 STEP 2 :: CALL KEY
(3,K,S):: IF S THEN 1930
ELSE CALL SOUND(-99,TQ+F,F)
:: CALL SOUND(-99,TQ+100+F,F
)
1920 NEXT F :: GOTO 1910
1930 CALL CLEAR :: RETURN
1940 !!!!!/PAUSE RUTINE!!!!
1950 SUB PAUSE(X):: FOR F=1
TO X :: NEXT F :: SUBEND
1960 !!!!!/
1970 END
```

Spectrum Økonomi



I programmet har du mulighed for 15 poster i hver måned og startes med GOTO 3720 den første gang, det køres. Dette gøres, for at få de danske karakterer genereret. Programmet har mulighed for at oprette budgetter, analyser, indtaste indtægt og se månedens forbrug. Som en ekstra finesse bliver budgetposterne automatisk sat i alfabetisk orden. □

Ole L. Larsen



HCT

```

1000 REM HJEMME-BUDGETM OLE L. LARSEN
1010 REM OPRETTE BUDGET
1020 DIM A(15,12)
1030 DIM A$(15,12,15)
1040 DATA "JANUAR", "FEBRUAR", "MARTS", "APRIL", "MAJ", "JUNI", "JULI", "AUGUST", "SEPTEMBER", "OKTOBER", "NOVEMBER", "DECEMBER"
1050 DIM M$(12,9)
1060 RESTORE
1070 FOR M=1 TO 12
1080 READ M$(M)
1090 NEXT M
1100 LET BI=0: LET BI1=0: LET BI2=0: LET BI3=0: LET BI4=0
1110 PRINT INK 1; INVERSE 1; AT 0,0; " BUDGET PROGRAM "
1120 PRINT INK 7; PAPER 1; AT 16,0; " DER ER PLADS TIL 15 POSTER "
1130 PRINT INK 7; PAPER 1; AT 17,0; " SKRIV 'X' I DEM DU IKKE BRUGER "
1140 BEEP .2,20: INPUT PAPER 6; "HVILKET HRSKKEK BUDGETTET "; Y
1150 FOR Q=1 TO 15
1160 PRINT AT Q,2-LEN STR$ Q; Q
1170 BEEP .2,20: INPUT PAPER 6; "NAVN PH UDGIFFTEN "; LINE Z$
1180 LET A$(Q,1)=Z$
1190 PRINT AT Q,3; A$(Q,1);
1200 NEXT Q
1210 CLS
1220 FOR Q=1 TO 15
1230 FOR M=1 TO 12
1240 PRINT INK 1; INVERSE 1; AT 0,0; "INDTAST BELFB FOR "
1250 PRINT OVER 1; FLASH 1; AT 0,17; A$(Q,1)
1260 PRINT INK 7; PAPER 1; AT 16,0; "SKRIV 0 I DE POSTER DER HEDDER X"
1270 PRINT AT M,2-LEN STR$ M; M; AT M,3; M$(M); " ";
1280 BEEP .2,20: INPUT PAPER 6; "BELFB FOR MHNEDEN "; A(Q,M)
1290 PRINT AT M,25-LEN STR$ A(Q,M); A(Q,M); AT M,27; "kr."
1300 NEXT M
1310 PRINT INK 7; PAPER 1; AT 18,0; " ER DISSE TAL RIGTIGE (J/N) "
1320 PAUSE 0
1330 IF INKEY$="J" AND Q<16 THEN CLS : NEXT Q
1340 IF INKEY$="J" AND Q>15 THEN GO TO 1370

```

```

1350 IF INKEY$="N" THEN CLS : GO TO 1230
1360 GO TO 1320
1370 GO SUB 3120
1380 GO TO 3330
1390 REM BUDGET OVERSIGT
1400 LET M=1
1410 CLS : PRINT INK 1; INVERSE 1; AT 0,0; " BUDGET OVERSIGT "
1420 PRINT OVER 1; FLASH 1; AT 0,18; M$(M)
1430 PRINT PAPER 6; AT 0,28; Y
1440 GO SUB 3230
1450 LET T=0
1460 FOR T=1 TO 15
1470 LET T=TO+AC(T,M)
1480 NEXT T
1490 PRINT BRIGHT 1; AT 17,2; "SAMLEDE UD GIFTER I "; M$(M)
1500 PRINT BRIGHT 1; AT 18,13; T; " kr."
1510 PRINT PAPER 6; BRIGHT 1; INK 0; AT 20,3; "5 = TILBAGE      8 = FREM"
1520 GO SUB 3300
1530 PAUSE 0
1540 IF INKEY$="0" THEN GO TO 3330
1550 IF INKEY$="5" THEN LET M=M-1: GO TO 0 1580
1560 IF INKEY$="8" THEN LET M=M+1: GO TO 0 1580
1570 GO TO 1530
1580 IF M<1 THEN LET M=1
1590 IF M>12 THEN LET M=12
1600 GO TO 1410
1610 REM RETTE BELFB I BUDGET
1620 CLS
1630 PRINT INK 1; INVERSE 1; AT 0,0; " RETTE BELFB I BUDGET "
1640 FOR Q=1 TO 15: PRINT AT Q,3-LEN STR$ Q; Q; PRINT AT Q,5; A$(Q,1); NEXT Q
1650 PRINT PAPER 1; INK 7; AT 17,0; "INDTAST >>0<< FOR MENTY "
1660 BEEP .2,20: INPUT PAPER 6; "INDTAST NUMMERET PH DEN BUDGET- POST DU VIL RETTE "; Q
1670 IF Q=0 THEN GO TO 3330
1680 CLS
1690 PRINT INK 1; INVERSE 1; AT 0,0; " RETTE BELFB I "
1700 PRINT OVER 1; FLASH 1; AT 0,16; A$(Q,1)
1710 FOR M=1 TO 12
1720 PRINT AT M,3-LEN STR$ M; M; PRINT AT M,5; M$(M); " ";
1730 PRINT AT M,25-LEN STR$ A(Q,M); A(Q,M); AT M,27; "kr."
1740 NEXT M
1750 PRINT PAPER 1; INK 7; AT 17,0; "INDTAST >>0<< FOR MENTY "
1760 BEEP .2,20: INPUT PAPER 6; "INDTAST NUMMERET PH MHNED DER SKAL RETTES "; M
1770 IF M=0 THEN GO TO 3330
1780 PRINT AT M,15; " "
1790 PRINT AT 17,0; " "
1800 BEEP .2,20: INPUT PAPER 6; "NYT BELFB "; A(Q,M)
1810 PRINT AT M,25-LEN STR$ A(Q,M); A(Q,M); AT M,27; "kr."
1820 PRINT PAPER 1; INK 7; AT 17,0; "INDTAST >>0<< FOR MENTY "
1830 GO TO 1690
1840 REM INDIESTE INDTSGT
1850 CLS
1860 PRINT INK 1; BRIGHT 1; AT 0,0; " INDTSGTER FOR HELE HRET "
1870 PRINT AT 2,0; "1 = BRUTTOINDTSGT "; AT 2,29; "kr."
1880 PRINT AT 4,0; "2 = HRLIG RENTEINDTSGT "; AT 4,29; "kr."
1890 PRINT AT 6,0; "3 = AKTIE-UDBYTTE "; AT 6,29; "kr."
1900 PRINT AT 8,0; "4 = ANDRE INDTSGTER "; AT 8,29; "kr."
1910 PRINT AT 10,0; "5 = SORT ARBEJDE "; AT 10,29; "kr."
1920 PRINT AT 2,28-LEN STR$ BI; BI
1930 PRINT AT 4,28-LEN STR$ BI1; BI1
1940 PRINT AT 6,28-LEN STR$ BI2; BI2
1950 PRINT AT 8,28-LEN STR$ BI3; BI3

```

```

1960 PRINT AT 10,28-LEN STR$ B14;B14
1970 PRINT AT 11,0;"
"
1980 PRINT PAPER 1; INK 7;AT 13,0;" TR
YK Ph DET TAL DU VIL RETTE "
1990 LET BIT=BI1+BI2+BI3+BI4
2000 PRINT AT 17,0;"hRLIG BRUTTO INDTsGT
";BIT;" kr."
2010 PRINT AT 19,0;" INDTsGT PR. MhNED
";INT (BIT/12);" kr."
2020 GO SUB 3300
2030 PAUSE 0
2040 IF INKEY$="0" THEN GO TO 3330
2050 IF INKEY$="1" THEN GO TO 2110
2060 IF INKEY$="2" THEN GO TO 2130
2070 IF INKEY$="3" THEN GO TO 2150
2080 IF INKEY$="4" THEN GO TO 2170
2090 IF INKEY$="5" THEN GO TO 2190
2100 GO TO 2030
2110 BEEP .2,20: INPUT PAPER 6;"hRLIG B
RUTTOINDTsGT ";B1
2120 GO TO 1840
2130 BEEP .2,20: INPUT PAPER 6;"hRLIG R
ENTE-INDTsGT ";B11
2140 GO TO 1840
2150 BEEP .2,20: INPUT PAPER 6;"AKTIE U
DBYTTE ";B12
2160 GO TO 1840
2170 BEEP .2,20: INPUT PAPER 6;"ANDRE I
NDTsGTER ";B13
2180 GO TO 1840
2190 BEEP .2,20: INPUT PAPER 6;"SORT AR
BEJDE ";B14
2200 GO TO 1840
2210 REM BUDGET ANALYSE
2220 LET SAM=0
2230 CLS
2240 LET X=0
2250 PRINT INK 1; INVERSE 1;AT 0,0;" BU
DGET-ANALYSE FOR HELE hRET "
2260 FOR Q=1 TO 15
2270 FOR M=1 TO 12
2280 LET X=X+A(Q,M)
2290 NEXT M
2300 PRINT AT Q,2-LEN STR$ Q;Q;AT Q,3;A$
(Q,1);"=";
2310 PRINT AT Q,25-LEN STR$ X;X;AT Q,27;
"kr."
2320 LET SAM=SAM+X
2330 LET X=0
2340 NEXT Q
2350 PRINT OVER 1;AT 15,0;"
"
2360 PRINT INK 3;AT 16,0;" SAMLEDE IND
TsGTER ";BIT;" kr."
2370 PRINT INK 3;AT 17,0;" INDTsGT PR.
MhNED ";INT (SAM/12);" kr."
2380 PRINT INK 2;AT 18,0;" SAMLEDE UDG
IFTER ";SAM;" kr."
2390 PRINT INK 2;AT 19,0;" MhNEDLIGE U
DGIFTER ";INT (SAM/12);" kr."
2400 PRINT INK 1;AT 20,0;"TIL FORBRUG "
;INT ((BI-SAM)/12);" kr. pr. MhNED"
2410 GO SUB 3300
2420 PAUSE 0
2430 IF INKEY$="0" THEN GO TO 3330
2440 GO TO 2420
2450 REM FJERNE BUDGETPOSTER
2460 CLS
2470 PRINT INK 1; INVERSE 1;AT 0,0;" FJ
ERNE BUDGETPOSTER "
2480 FOR Q=1 TO 15
2490 PRINT AT Q,2-LEN STR$ Q;Q;AT Q,3;A$
(Q,1)
2500 NEXT Q
2510 GO SUB 3300
2520 PRINT INK 7; PAPER 1;AT 16,0;"HUIL
KEN BUDGETPOST VIL DU FJERNE"
2530 BEEP .2,20: INPUT Q
2540 IF Q=0 THEN GO TO 3330
2550 LET A$(Q,1)="X
2560 FOR M=1 TO 12
2570 LET A(Q,M)=0
2580 NEXT M
2590 GO SUB 3120
2600 CLS
2610 PRINT INK 1; INVERSE 1;AT 0,0;" FJ
ERNE BUDGETPOSTER
2620 FOR Q=1 TO 15
2630 PRINT AT Q,2-LEN STR$ Q;Q;AT Q,3;A$
(Q,1)
2640 NEXT Q
2650 BEEP .2,20: PRINT INK 7; PAPER 1;A
T 16,0;"VIL DU LAVE EN NY BUDGETPOST J/N
"
2660 PAUSE 0
2670 IF INKEY$="N" THEN GO TO 2700
2680 IF INKEY$="J" THEN GO TO 2760
2690 GO TO 2660
2700 BEEP .2,20: PRINT INK 7; PAPER 1;A
T 16,0;"VIL DU FJERNE FLERE BUDGETPOSTER
"
2710 PRINT INK 7; PAPER 1;AT 17,14;"(J/
N)"
2720 PAUSE 0
2730 IF INKEY$="N" THEN GO TO 3330
2740 IF INKEY$="J" THEN GO TO 2450
2750 GO TO 2720
2760 REM OPRETTE NY BUDGETPOST
2770 CLS
2780 PRINT INK 1; INVERSE 1;AT 0,0;" OP
RETTE NY BUDGETPOST "
2790 FOR Q=1 TO 15
2800 PRINT AT Q,2-LEN STR$ Q;Q;AT Q,3;A$
(Q,1)
2810 NEXT Q
2820 PRINT OVER 1;AT 15,0;"
"
2830 FOR Q=1 TO 15
2840 IF A$(Q,1)="X " THEN
GO TO 2910
2850 NEXT Q
2860 BEEP .2,20: PRINT INK 7; PAPER 1;A
T 16,0;"DER ER IKKE PLADS TIL EN NY POST
"
2870 GO SUB 3300
2880 PAUSE 0
2890 IF INKEY$="0" THEN GO TO 3330
2900 GO TO 2880
2910 BEEP .2,20: INPUT PAPER 6;"NAUN TI
L BUDGETPOST ? "; LINE Z$
2920 LET A$(Q,1)=Z$
2930 CLS
2940 PRINT INK 1; INVERSE 1;AT 0,0;"IND
TAST BELfB I "
2950 PRINT OVER 1; FLASH 1;AT 0,16;A$(Q
,1)
2960 FOR M=1 TO 12
2970 PRINT AT M,3-LEN STR$ M;M: PRINT AT
M,5;M*(M);"=";
2980 PRINT AT M,25-LEN STR$ A(Q,M);A(Q,M)
;AT M,27;"kr."
2990 NEXT M
3000 FOR M=1 TO 12
3010 PRINT AT M,15;" "
3020 BEEP .2,20: INPUT PAPER 6;"NYT BEL
fB ";A(Q,M)
3030 PRINT AT M,25-LEN STR$ A(Q,M);A(Q,M)
;AT M,27;"kr."
3040 NEXT M
3050 GO SUB 3120
3060 BEEP .2,20: PRINT INK 7; PAPER 1;A
T 17,0;" VIL DU LAVE FLERE BUDGETPOSTER
"
3070 PRINT INK 7; PAPER 1;AT 18,14;"(J/
N)"
3080 PAUSE 0
3090 IF INKEY$="N" THEN GO TO 3330
3100 IF INKEY$="J" THEN GO TO 2760
3110 GO TO 3080
3120 REM ALFABETISK ORDEN
3130 PRINT AT 17,0;" VENT LIGE ET fJ
EBLIK "
3140 FOR K=1 TO 15: FOR C=K+1 TO 15
3150 IF A$(K,1)<=A$(C,1) THEN GO TO 320
0
3160 FOR M=1 TO 12
3170 LET E=A$(K,M): LET A$(K,M)=A$(C,M)
: LET A$(C,M)=E$
3180 LET E=A$(K,M): LET A(K,M)=A(C,M): LE
T A(C,M)=E
3190 NEXT M
3200 NEXT C
3210 NEXT K
3220 RETURN
3230 REM OPSTILLING
3240 FOR Q=1 TO 15
3250 PRINT AT Q,2-LEN STR$ Q;Q;AT Q,3;A$
(Q,1);"=";
3260 PRINT AT Q,25-LEN STR$ A(Q,M);A(Q,M)
;AT Q,27;"kr."
3270 NEXT Q
3280 PRINT OVER 1;AT 15,0;"
"
3290 RETURN
3300 REM TRYK 0 FOR MENY
3310 PRINT INK 0; PAPER 6;AT 21,3;"TRYK
Ph >>0<<TAST FOR MENY"
3320 RETURN
3330 REM MENY
3340 BORDER 4
3350 CLS
3360 POKE 23609,20
3370 POKE 23658,8
3380 PRINT INVERSE 1;AT 0,0;" MENY ME
NY MENY MENY MENY "
3390 PRINT "' 1 = LAVE NYT BUDGET"
3400 PRINT "' 2 = BUDGET OVERSIGT"
3410 PRINT "' 3 = BUDGET ANALYSE"
3420 PRINT "' 4 = INDTASTE INDTsGTER"
3430 PRINT "' 5 = RETTE BELfB I BUDGET"
3440 PRINT "' 6 = FJERNE BUDGET POST"
3450 PRINT "' 7 = OPRETTE NY BUDGET POST
"
3460 PRINT "' 8 = SAVE DETTE PROGRAM MED
DATA" "
3470 PRINT PAPER 7; INK 2; FLASH 1;'"
UED BREAK BRUG GOTO 3330 "
3480 PAUSE 0
3490 IF INKEY$="1" THEN GO TO 3580
3500 IF INKEY$="2" THEN GO TO 1390
3510 IF INKEY$="3" THEN GO TO 2210
3520 IF INKEY$="4" THEN GO TO 1840
3530 IF INKEY$="5" THEN GO TO 1610
3540 IF INKEY$="6" THEN GO TO 2450
3550 IF INKEY$="7" THEN GO TO 2760
3560 IF INKEY$="8" THEN GO TO 3650
3570 GO TO 3480
3580 BEEP .2,10: BEEP .2,20: BEEP .2,30:
BEEP .2,40: BEEP .2,50: PRINT BRIGHT 1
;AT 18,0;" ##### ER DU HELT SIKKER?. ###
# "
3590 PRINT BRIGHT 1;" DU SLETTER ALLE D
E DATA DER ER "
3600 PRINT BRIGHT 1;" I BUDGETTET NU.
TRYK (J/N) "
3610 PAUSE 0
3620 IF INKEY$="J" THEN RUN
3630 IF INKEY$="N" THEN GO TO 3330
3640 GO TO 3610
3650 REM SAVE ROUTINE
3660 BEEP .5,0
3670 CLS : PRINT AT 10,0;" GfR KLAR
TIL OPTAGELSE "
3680 SAVE "BUDGET" LINE 3720
3690 PRINT AT 10,0;" SPOL TILBAGE FOR
VERIFY "
3700 BEEP 1,0: VERIFY "BUDGET"
3710 STOP
3720 REM DANSKE BOGSTAVER
3730 RESTORE 3750: FOR X=0 TO 7: READ S
3740 POKE USR "S"+X,S
3750 DATA 0,30,48,80,126,80,94,0
3760 NEXT X
3770 RESTORE 3790: FOR X=0 TO 7: READ F
3780 POKE USR "F"+X,F
3790 DATA 0,60,70,74,82,98,60,0
3800 NEXT X
3810 RESTORE 3830: FOR X=0 TO 7: READ H
3820 POKE USR "H"+X,H
3830 DATA 24,0,60,66,66,126,66,0
3840 NEXT X
3850 GO TO 3330
3860 REM
BYTES FORBRUG
3870 PRINT "' Der er brugt ";(PEEK 236
41+256*PEEK 23642)-(PEEK 23635+256*PEEK
23636);" bytes: PRINT "' Der er ";6553
6-USR 7962;" bytes tilbage": LET mer=655
36-USR 7962: LET qmer=mer/1024: PRINT "'
Der er ca. ";(INT (qmer*100))/100;" K
tilbage"
3880 STOP

```



Programmet består af to subrutiner og har til formål at reducere længden af strenge, idet den første subrutine kan reducere antallet af karakterer i en vilkårlig streng med 1 for hver fjerde karakter. Den anden subrutine kan så til gengæld genskabe den oprindelige streng i fuld længde.

Programmet vil specielt være anvendeligt i databaseprogrammer, hvor det kan give kapacitetsforøgelse, som svarer til typisk 5-10 K, ekstra hukommelse (48 K spectrum). Der er imidlertid også nogle begrænsninger:

Stregene må kun rumme bogstaver (store som små), tal samt punktum og mellemrum.

Både kodning og dekodning tager lidt tid, men programmet er så kompakt opbygget, at det ikke er overvældende. F.eks. vil det for en streng på 30 karakterer tage ca. 2 sek. I en database bør man ikke anvende programmet på de oplysninger, man typisk vil søge, under (f.eks. journalnumre, personnummer, navne ell. lign.).

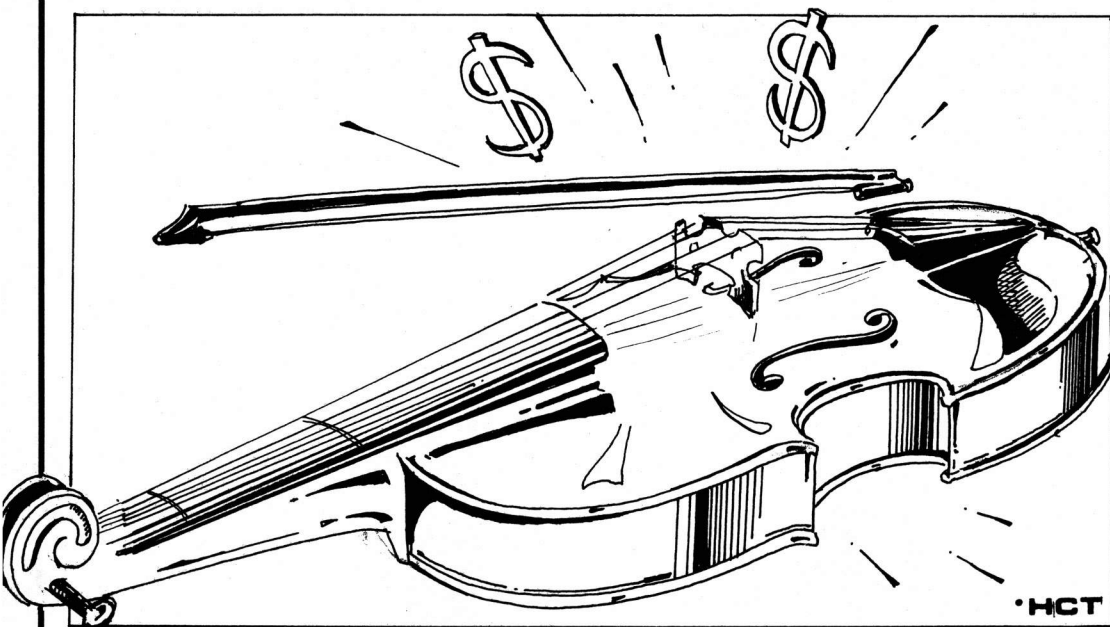
Disse oplysninger bør man bevare i ukodede strenge.

I mange programmer vil disse ting ikke have så stor betydning sammenlignet med den kapacitetsforøgelse man vil få til rådighed.

I øvrigt kan oplyses, at for begge subrutiner gælder, at de ændrer et x til y. Det y, der kommer ud af første rutine, er således kodet, mens det y, som kommer ud af 2. rutine, er dekodet igen.

Strenge af enhver længde kan behandles, programmet skulle være helt problemløst (og i 100% BASIC). Det kan derfor uden større problemer indlægges i eksisterende programmer. □

Michael Gravensen



Spectrum strenge

4 REM Kodningsprocedure

```

10 LET L=LEN X$: LET st=4*(INT
(L/4))-3: LET Y$="": IF L<3 THE
N LET Y$=X$: RETURN
15 LET v0=0: DIM x(4): IF st+6
=L THEN LET X$=X$+" ": GO TO 10
20 FOR r=1 TO st STEP 4: FOR n
=1 TO 4: LET xa=CODE X$(r+n-1):
LET x(n)=xa-46-7*(xa>=65)-6*(xa>
=97)+15*(xa=32): IF x(n)>=64 OR
X(n)<0 THEN GO TO 9999
25 NEXT n: LET q1=INT (x(4)/16
): LET Y$=Y$+CHR$(x(1)+64*q1):
LET q2=INT ((x(4)-16*q1)/4): LET
Y$=Y$+CHR$(x(2)+64*q2): LET Y$
=Y$+CHR$(x(3)+64*(x(4)-4*(4*q1+
q2))): LET v0=v0+1: NEXT r: LET
Y$=Y$+X$(st+4 TO L): RETURN

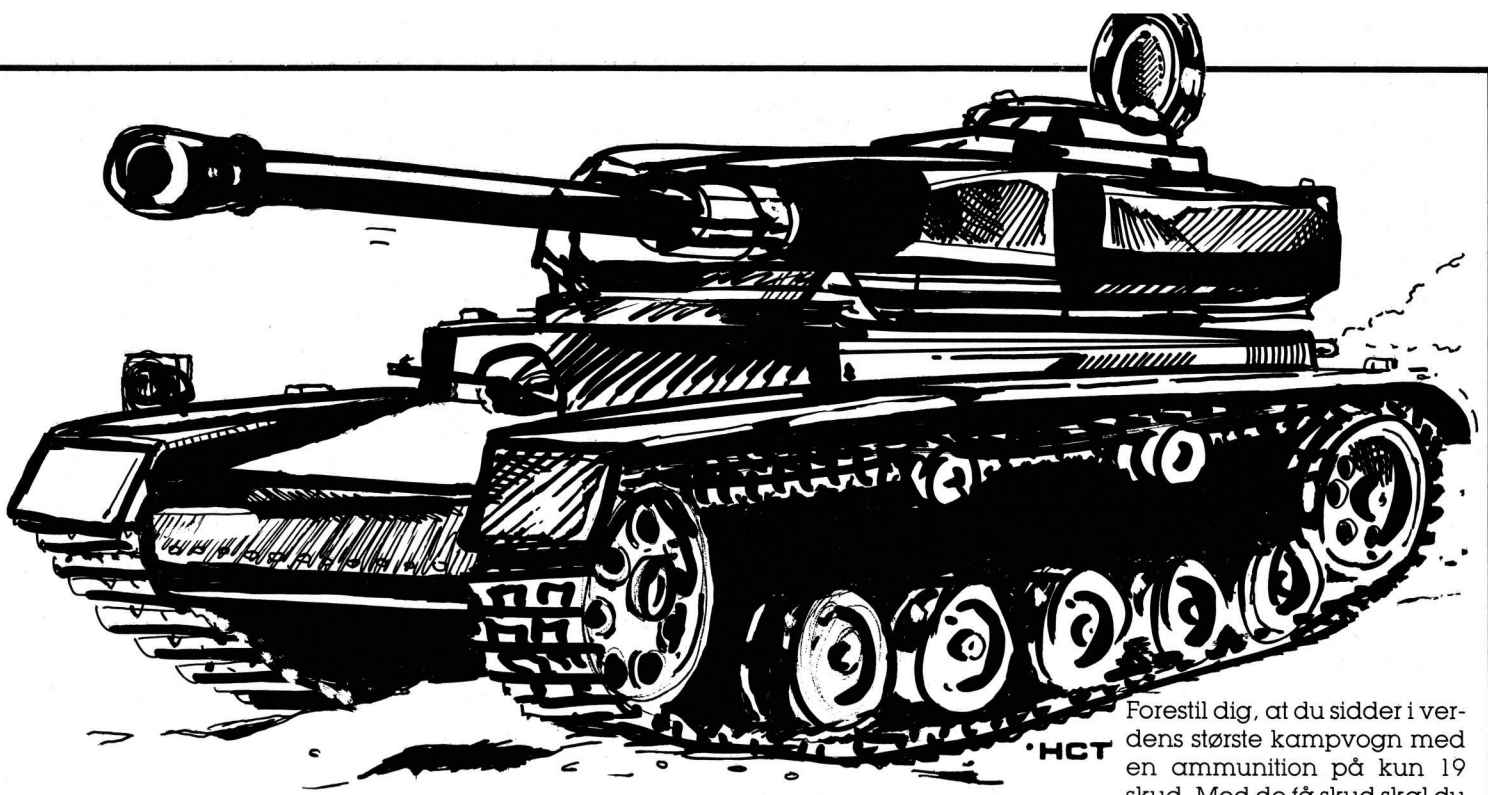
```

50 REM Udprintningsprocedure

```

55 LET L=LEN X$: DIM x(4): DIM
y(4): LET Y$="": IF L<3 THEN LE
T Y$=X$: RETURN
60 FOR r=1 TO L-2 STEP 3: FOR
n=1 TO 3: LET x(n)=CODE X$(r+n-1
): LET y(n)=INT (x(n)/64): LET x
(n)=x(n)-64*y(n): NEXT n: LET x(
4)=y(1)*16+y(2)*4+y(3): FOR n=1
TO 4: LET Y$=Y$+CHR$(x(n)+46+7*
(x(n)>=12)+6*(x(n)>37)-15*(x(n)=
1)): NEXT n: NEXT r: LET Y$=Y$+X
$(r TO L): RETURN

```



'HCT

Forestil dig, at du sidder i verdens største kampvogn med en ammunition på kun 19 skud. Med de få skud skal du forsvare dig mod de fjendtlige rumskibe.

Rumskibene flyver i forskellige højder, og pointfordelingen er også sat derefter. Det drejer sig med andre ord om at nedskyde rumskibene i den største højde, det er muligt.

Din kanon affyres ved at anvende et joystick eller trykke på U-tasten. □

Claus Evers Christensen

Multiaction

```

1 REM C.E.CHRISTENSEN
2 PRINT AT 0,0; "
3 PRINT AT 10,9; "SKYD FJENDE"
4 PRINT AT 12,7; "PÅRES U TIL 3"
5 PRINT AT 21,0; "
6 IF INKEY$="" THEN GOTO 6
7 PRINT AT 10,9; "
8 FOR A=14 TO 25
9 PRINT AT 19,A-1; "
10 PRINT AT 20,A-2; "
11 SC=0
17 PRINT AT 19,A; "
18 PRINT AT 20,A-1; "
19 NEXT A
20 A=0
21 PRINT AT 19,0; "SKUD = ";A
22 H=INT (RND*10)+4
23 PRINT AT 17,0; "SCORE = ";SC
24 PAUSE 20
40 G=2
50 PRINT AT H,G; "▲"
60 PAUSE 3
70 PRINT AT H,G; "
75 IF INKEY$="U" THEN GOTO 500
80 G=G+1
90 IF G=30 THEN GOTO 22
95 GOTO 50
98 A=A+1
99 FOR K=10 TO 15
100 SOUND K,50
101 NEXT K
102 B=10
106 PRINT AT H,G-1; "
107 PRINT AT H,G; "▲"
108 PRINT AT B,26; "▲"
109 PAUSE 1
110 PRINT AT B,26; "
115 PAUSE 3

```

```

550 B=B-1
560 G=G+1
580 IF B<1 THEN LET B=1
581 PRINT AT H,26; "
586 IF G>29 THEN GOTO 21
587 IF PEEK B=PEEK H AND PEEK G
=PEEK 26 THEN GOTO 1000
590 IF A<20 THEN GOTO 516
595 GOTO 1140
1000 PRINT AT H-1,G-1; "
1010 PAUSE 5
1020 PRINT AT H-2,G-2; "
1030 PAUSE 5
1040 PRINT AT H-2,G-2; "
1050 PAUSE 5
1060 PRINT AT H-1,G-1; "
1070 FAST
1080 FOR X=20 TO 40
1090 SOUND X,100
1100 NEXT X
1110 SLOW
1120 LET SC=SC+20-H
1122 PRINT AT 17,0; "SCORE = ";SC
1125 PRINT AT 19,0; "SKUD = ";A
1130 IF A<20 THEN GOTO 22
1140 PRINT AT 11,10; "GAME OVER"
1150 PAUSE 250
1160 CLS
1170 GOTO 1
1300 SAVE "SKYD FJENDE"
1310 RUN

```



House Defender

Når du har startet programmet, der passer til en VIC-20, kommer en lille UFO flyvende hen over skærmen. Du skal

skyde den ned, inden den når frem til din egen kanon eller før den får skudt dit hus helt i grus.

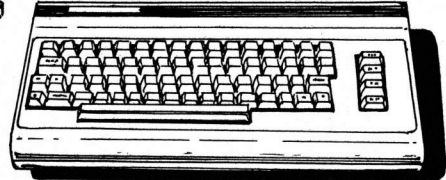
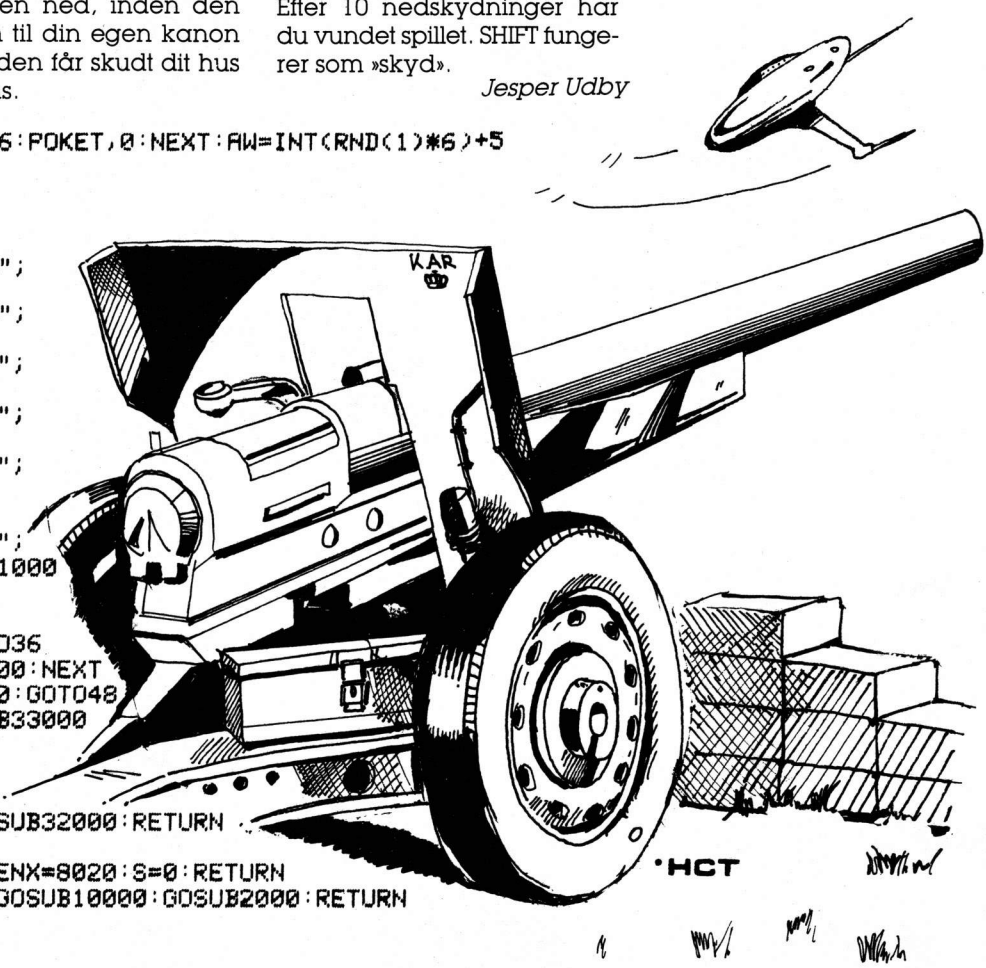
Efter 10 nedskydninger har du vundet spillet. SHIFT fungerer som »skyd».

Jesper Udby

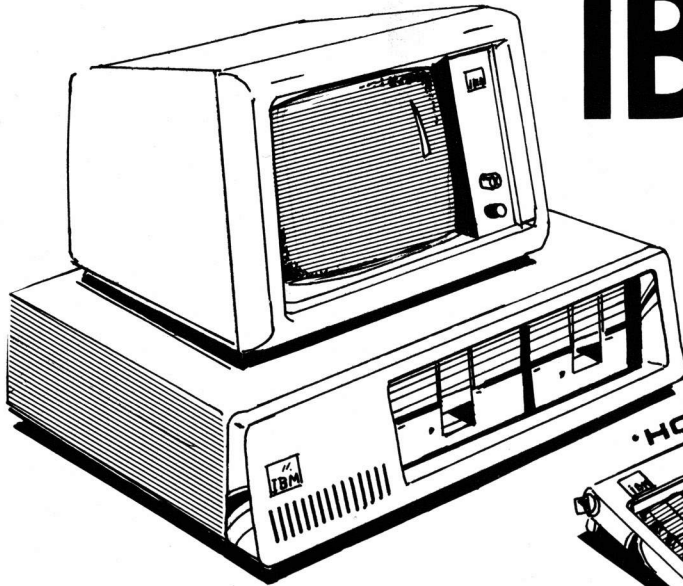
```

0 PRINT "J": FORT=38400T038400+506: POKET,0: NEXT: AW=INT(RND(1)*6)+5
1 POKE36879,PEEK(36879)AND247
2 PRINT "W;WQ
10 PRINT "XXXXXXXXXX"
12 PRINT "
14 PRINT "
15 PRINT "
16 PRINT "
18 PRINT "
20 PRINT "
22 PRINT "
24 PRINT "
26 PRINT "
28 PRINT "
30 PRINT "
32 PRINT "
34 PRINT "
36 IFPEEK(653)=1ANDS=0THENGOSUB1000
37 IFS=1THENGOSUB1002
38 GOSUB30000
40 IFS=0THENGOSUB1010:SS=1:GOTO36
41 IFS=0THENFORXX=1TO6:GOSUB20000:NEXT
44 POKEL,32:L=L+1:IFL=LLTHENSS=0:GOTO48
46 POKEL,61:IFL=BTTHENBS=1:GOSUB33000
48 IFS=1THENGOSUB3000
50 GOTO36
999 GOTO36
1000 S=1:POKE8020,160:X=8020:GOSUB32000:RETURN
1002 FORQ=1TO2
1003 POKEX,32:X=X-22:IFX<7680THENX=8020:S=0:RETURN
1004 IFPEEK(X)=61THENGOSUB6010:GOSUB10000:GOSUB2000:RETURN
1006 POKEX,46:NEXT
1008 RETURN
1010 LX=RND(1)*6
1012 L=7680+LX*22+44:LL=L+22
1014 LX=RND(1)*200+100
1016 LX=RND(1)*15:BT=L+LX-1
1017 IFWQ/AW=INT(WQ/AW)ANDWQ<0THENBT=7883:L=7878:LL=7891:AW=AW-INT(RND(1)*2)
1018 POKEL,160:RETURN
2000 W=W+1:PRINT "W;WQ:SS=0:S=0:BS=0:IFW=10THENPRINT "DU VANDT":GOTO2006
2001 POKEX-1,32:POKEX,32:POKEX+1,32:POKEL,32:POKEBT,32
2002 POKEX-1,32:POKEX,32:POKEX+1,32:POKEL,32:POKEBT,32
2004 RETURN
2006 IFWQ>20THENPRINT "MEN DINE HUSE ER UBRU-GELIGE"
2008 END
3000 POKEBT,32:BT=BT+23:IFPEEK(BT)<>32ANDPEEK(BT)<>46THENGOSUB6000
3001 IFXW=1THENGOSUB10000:GOSUB4000:XW=0:RETURN
3002 IFBT>8185THENBS=0:RETURN
3004 POKEBT,35:RETURN
4000 WQ=WQ+1:BS=0:POKEBT,32:PRINT "W;WQ:IFWQ=30THENEND
4002 RETURN
6000 IFPEEK(BT)=102ORPEEK(BT)=104THENXW=1:POKEBT,86:RETURN
6002 IFPEEK(BT)=35THENRETURN
6004 GOSUB10000:POKE36879,PEEK(36879)OR8:GOSUB10000:POKE36879,PEEK(36879)AND247
6006 GOSUB4000:PRINT "DU HAR TABT.":PRINT "DIN KANON ER ØDELAGT!!"
6009 END
6010 POKEX-1,160:POKEX,160:POKEX+1,160:IFBS=1THENPOKEBT,87
6012 RETURN
10000 REM EXPLOSION
10003 POKE36877,130
10006 FORLQ=15TO0STEP-.8:POKE36878,LQ:FORMQ=1TO150:NEXTMQ,LQ
10009 POKE36877,0:POKE36878,0:RETURN
20000 GOTO21000
21000 RETURN
30000 XW=XW+1:IFXW/3=INT(XW/3)THENPOKE36876,200:POKE36878,15:FORTY=1TO9:NEXT:POK
E36876,0
31000 RETURN
32000 POKE36876,190:POKELS,15:F0RET=1TO10:NEXT:POKE36876,0:RETURN
33000 POKE36876,220:POKE36878,15:F0RET=1TO15:NEXT:POKE36876,0:RETURN

```

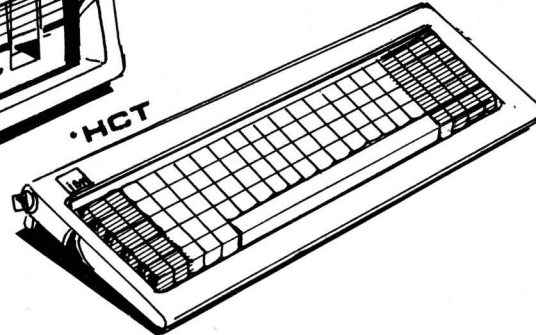


IBM-Days



Programmet er selvforklarende

Kurt Friis Hansen



```
100 REM TEST GYLDIG DATO - linje 120-160 kan benyttes til test af indtastning
110 REM
120 GOSUB 400
130 INPUT "INDTAST DATO,MANED,AR";DD%,MM%,YY%
140 PRINT DD%
150 GOSUB 490
160 PRINT DD%:GOTO 130
170 '
180 ' Check af korrekt antal dage i måneden
190 '
200 ' Programmet tager højde for skudår, også at der kun er skudår ved
210 ' århundredskift, når årstallet er deleligt med fire hundrede.
230 ' Rutinen kræver at:
240 '           YY% indeholder årstal - f.eks. 1984
250 '           MM% indeholder måned - f.eks. 12
260 '           DD% indeholder dato - f.eks. 31
270 '
280 ' Rutinen kaldes som subroutine, og returnerer værdien -1, hvis datoen
290 ' i den specificerede måned er ulovlig (evt. kun det pågældende år).
300 '
310 ' Rutinen kender kun den Gregorianske kalender, der er gyldig fra
320 ' 15. oktober 1582. Datoen - variabelen DD% - returneres som -1, hvis en
330 ' tidligere dato specificeres.
340 '
350 ' Øvrige anvendte variabelnavne: INGEN
360 ' Anvendte funktionsnavne      : GYLDIG
370 '
380 ' Placeres tidligt i programmet:
390 '
400 DEF FN GYLDIG=31+((MM%=4)OR(MM%=6)OR(MM%=9)OR(MM%=11))-((MM%=2)*((O=(YY% MOD
410 ' 100))-O=(YY% MOD 4))-O=(YY% MOD 400))-3)
420 ' Bemærk anvendelsen af sammenligninger og relationer! Er en betingelse
430 ' opfyldt returneres værdien '-1' ellers værdien '0' (nul)!
440 '
450 ' Det følgende er den egentlige subroutine, som kan placeres frit i et
460 ' program. Placeres subrutiner og funktioner tidligt i programmet, sikres
470 ' maksimal arbejdshastighed!
480 '
490 IF YY%<1583 THEN GOSUB 530
500 IF DD%<1 OR (DD%>FN GYLDIG) THEN DD%=-1
510 IF MM%<1 OR MM%>12 THEN DD%=-1
520 RETURN
530 IF YY%<1582 OR MM%<10 THEN 560
540 IF MM%>10 THEN RETURN
550 IF DD%>14 THEN RETURN
560 DD%=-1
570 RETURN 520
580 '
590 ' Bemærk brugen af den specielle RETURN 520 kommando, som
600 ' i realiteten svarer til en GOTO 520, blot med den forskel,
610 ' at RETUR-adressen fjernes fra computerens STACK.
```

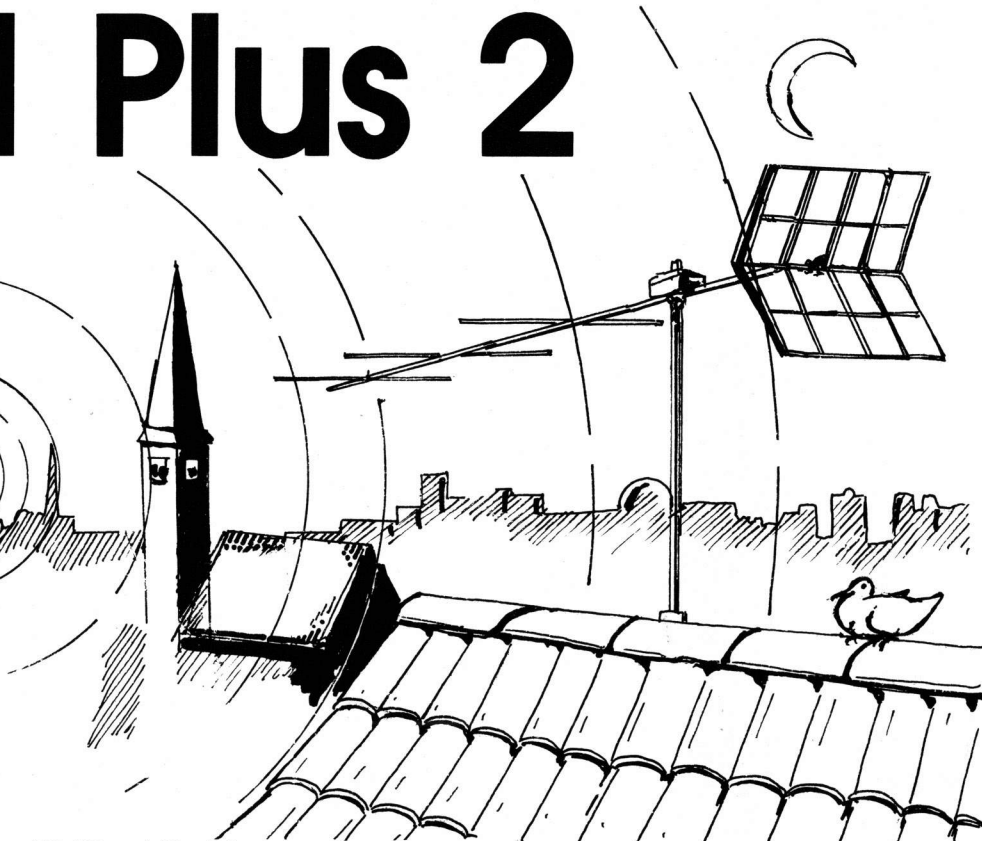
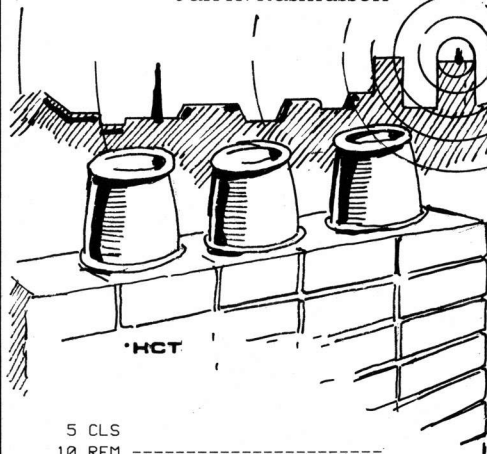
**IBM-
PC**

Kanal Plus 2

Det vrirler nu med lokale tv-stationer og mange sparer penge ved at lave deres antenner selv.

Med dette Spectrum-program kan du nu selv beregne din antenne.

Jan H. Rasmussen



```

5 CLS
10 REM -----
15 REM ANTENNE BEREGNINGS PROG
20 REM
25 REM JAN H. RASMUSSEN, AABYHOJ
30 REM -----
40 DIM d(10)
45 DIM a(10)
50 PRINT "VALG AF ANTAL ANTENNE ELEMEN
TER": PLOT 0,164: DRAW 255,0
55 PRINT
60 PRINT "Du kan vaelge et antal melle
m", "4 og 10"
80 INPUT "Antal elementer ";el
85 IF el<4 OR el>10 THEN GO TO 80
90 PRINT ", "Jeg vaelger "; FLASH 1;el;
FLASH 0;" elementer"
95 PRINT ", "Skal det vaere en UHF elle
r en UHF antenne"
100 PRINT ", "UHF tast 1", "UHF tast 2"
110 INPUT f
120 IF f<1 OR f>2 THEN GO TO 110
130 GO SUB f*1000
150 REM -----
151 REM REFLEKTOR UDREGNING
152 REM -----
155 LET ref=(150/frq)*1000
160 REM -----
161 REM DIPOL UDREGNING
162 REM -----
165 LET dip=(72.5/frq)*1000
170 REM -----
171 REM DIREKTOR UDREGNING
172 REM -----
175 LET d(1)=(145/frq*1000)-(7250/frq)
180 FOR n=2 TO el-2
185 LET d(n)=d(n-1)-(d(n-1)/100*5)
190 NEXT n
200 REM -----
201 REM ELEMENT AFSTAND
202 REM -----
205 LET agr=300/frq*250
210 FOR n=1 TO 3: LET a(n)=agr: NEXT n

```

```

220 FOR n=4 TO el-1
230 LET a(n)=a(n-1)+(a(n-1)/100*4)
235 NEXT n
250 REM -----
251 REM LAENGDE AF HOVED BOM
252 REM -----
255 LET t=0
260 FOR n=1 TO el-1
265 LET t=t+a(n)
270 NEXT n
275 LET lsum=INT (t+40)
300 REM -----
301 REM UDSKRIFT PAA SKAERNEN
302 REM -----
305 CLS
310 PRINT AT 0,3;"UHF-UHF ANTENNE BEREG
NING"
355 PLOT 9,128: DRAW 143,0
360 PLOT 9,127: DRAW 143,0
365 PLOT 11,114: DRAW 0,28: PLOT 12,114
: DRAW 0,28
370 PLOT 25,115: DRAW 0,10: PLOT 25,131
: DRAW 0,10: PLOT 26,115: DRAW 0,10: PLO
T 26,131: DRAW 0,10
375 PLOT 39,116: DRAW 0,24: PLOT 40,116
: DRAW 0,24
380 PLOT 53,116: DRAW 0,24: PLOT 54,116
: DRAW 0,24
385 PLOT 68,117: DRAW 0,22: PLOT 69,117
: DRAW 0,22
390 PLOT 83,117: DRAW 0,22: PLOT 84,117
: DRAW 0,22
395 PLOT 99,118: DRAW 0,20: PLOT 100,11
8: DRAW 0,20
400 PLOT 115,118: DRAW 0,20: PLOT 116,1
18: DRAW 0,20
405 PLOT 132,119: DRAW 0,18: PLOT 133,1
19: DRAW 0,18
410 PLOT 149,120: DRAW 0,16: PLOT 150,1
20: DRAW 0,16
420 PRINT AT 2,3;"Reflektor"
425 PLOT 23,152: DRAW -11,-8
430 PRINT AT 3,10;"Direktore"
440 PRINT AT 8,5;"Dipol (isoleret fra a
nt.)"
445 PLOT 39,108: DRAW -13,5
450 PRINT AT 5,21;"Mod sender"
455 PLOT 168,124: DRAW 80,0
458 PLOT 246,126: DRAW 2,-2: DRAW -2,-2
460 PRINT AT 2,22;"Hovedbom"
470 PLOT 175,152: DRAW -21,-24
500 PRINT INVERSE 1;AT 9,0;"Antennen s
kal have flg. laengder"

```

```

520 PRINT AT 11,0;"Hovedbom ";lsum;AT
12,0;"Reflektor ";INT ref
530 PRINT AT 13,0;"Hver dipol ";INT dip
;AT 14,0;"1.direktor ";INT d(1)
540 PRINT AT 15,0;"2.direktor ";INT d(2
)
550 FOR n=3 TO el-2
560 PRINT AT 8+n,17;n;" .direktor ";INT
d(n)
570 NEXT n
580 PRINT "FLASH 1;AT 16,0;" Alle maal
i mm "
600 PRINT INVERSE 1;AT 17,0;" Afstand
e mellem elementerne "
620 FOR n=1 TO 3
625 PRINT AT 17+n,0;"M1";n;"&";n+1;" ";
INT a(n)
630 NEXT n
650 FOR n=4 TO 6
655 PRINT AT 14+n,11;"M1";n;"&";n+1;" "
;INT a(n)
660 NEXT n
670 FOR n=7 TO 9
675 PRINT AT 11+n,22;"M1";n;"&";n+1;" "
;INT a(n)
680 NEXT n
700 PRINT INVERSE 1;AT 21,0;"TAST FOR
COPY=1 NY ANT.=2 SAVE=3"
710 INPUT s
720 IF s=1 THEN COPY
730 IF s=2 THEN RUN
740 IF s=3 THEN GO TO 3000
750 GO TO 710
1000 PRINT ", "Du har valgt en UHF antenn
e"
1010 PRINT ", "Hvad for en TU kanal skal
den bruges til (21-69)"
1020 INPUT "Kanal ";juk
1030 IF uk<21 OR uk>69 THEN GO TO 1020
1040 LET frq=((uk-21)*8)+472
1050 RETURN
2000 PRINT ", "Du har valgt en UHF antenn
e"
2010 PRINT ", "Hvad for en TU kanal skal
den bruges til (2-12)"
2020 INPUT "Kanal ";juk
2030 IF uk=2 OR uk=3 OR uk=4 THEN LET f
rq=((uk-2)*7)+49: RETURN
2040 IF uk<5 OR uk>12 THEN GO TO 2020
2045 LET frq=((uk-5)*7)+176
2050 RETURN
3000 SAVE "ANTENNE" LINE 1

```



Happy Days

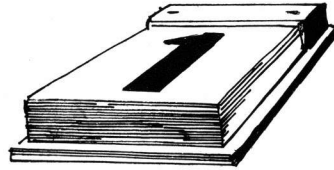
Hvor mange dage har du levet, og hvilken dag blev du egentlig født? Alt dette og ikke mere, erfarer du ved hjælp af dette lille Vic-20 program.

Niels-Jørgen Bertelsen

```

30 PRINT "PROGRAMMET BEREGNER HVOR MANGE DAGE DU HAR LEVET, OG HVILKEN DAG"
35 PRINT "DU ER FØDT PAA."
40 PRINT
45 INPUT "DATO(NU) :";A
50 IF A<1 THEN PRINT "I#####":GOTO45
55 INPUT "MAAED(NU) :";B
60 IF B<1 THEN PRINT "I#####":GOTO55
65 INPUT "AAR(NU) :";C
70 IF C<1 THEN PRINT "I#####":GOTO65
75 PRINT:INPUT "UGEDAG(NU) :";U$
80 IF U$="" THEN PRINT "I#####":GOTO75
85 PRINT:INPUT "F.DATO :";D
90 IF D<1 THEN PRINT "I#####":GOTO85
95 INPUT "F.MAANED :";E
100 IF E<1 THEN PRINT "I#####":GOTO95
105 INPUT "F.AAR :";F
110 IF F<1 THEN PRINT "I#####":GOTO105
115 IF U$="MANDAG" THEN U=1
120 IF U$="TIRSDAG" THEN U=2
125 IF U$="ONSDAG" THEN U=3
130 IF U$="TORSDAG" THEN U=4
135 IF U$="FREDAG" THEN U=5
140 IF U$="LØRDAG" THEN U=6
145 IF U$="SØNDAG" THEN U=7
146 IF C<F OR (C=F AND B<E AND A<D) THEN PRINT "IIIIII#####":GOTO85
147 G=0
150 IF C=F THEN A05
155 GOSUB375
160 YD=I-D
165 IF E=12 THEN I190
170 FORG=E+1 TO 12
175 GOSUB375
180 YD=YD+I
185 NEXT
190 YD=YD+((C-F-1)*365)+INT((C-F)/4)
195 IF B=1 THEN I220
200 FORG=1 TO B-1
205 GOSUB375
210 YD=YD+I

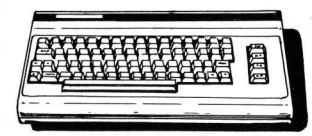
```



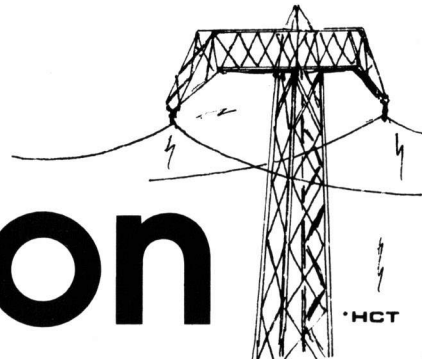
```

215 NEXTG
220 YD=YD+A
225 IFF/4<>INT(F/4) AND E<3 THEN YD=YD-1
230 IFF/4=INT(F/4) AND E>2 THEN YD=YD-1
235 PRINT
240 IF YD=1 THEN PRINT "LEVET I "YD" DAG.":GOTO260
245 PRINT "LEVET I "YD" DAGE."
260 IF YD/7=INT(YD/7) THEN I290
265 YD=YD-(INT(YD/7)*7)
270 FOR J=YD TO 1 STEP -1
275 U=U-1
280 IF U<1 THEN U=7
285 NEXT J
290 IF U=1 THEN Q$="MANDAG"
295 IF U=2 THEN Q$="TIRSDAG"
300 IF U=3 THEN Q$="ONSDAG"
305 IF U=4 THEN Q$="TORSDAG"
310 IF U=5 THEN Q$="FREDAG"
315 IF U=6 THEN Q$="LØRDAG"
320 IF U=7 THEN Q$="SØNDAG"
325 PRINT
330 PRINT "FØDT PAA EN "Q$"."
335 PRINT
340 PRINT "RUN/STOP (R/S)"
345 GET R$:IF R$="" THEN I345
346 IF R$<"R" THEN PRINT "J":END
350 D=0:E=0:F=0:YD=0:PRINT "J":GOTO85
375 IFG=10RG=30RG=50RG=70RG=80RG=100RG=120RG=140RG=160RG=180RG=200RG=220RG=240RG=260RG=280RG=300RG=320RG=340RG=360RG=380RG=400RG=420RG=440RG=460RG=480RG=500RG=520RG=540RG=560RG=580RG=600RG=620RG=640RG=660RG=680RG=700RG=720RG=740RG=760RG=780RG=800RG=820RG=840RG=860RG=880RG=900RG=920RG=940RG=960RG=980RG=1000RG=1020RG=1040RG=1060RG=1080RG=1100RG=1120RG=1140RG=1160RG=1180RG=1200RG=1220RG=1240RG=1260RG=1280RG=1300RG=1320RG=1340RG=1360RG=1380RG=1400RG=1420RG=1440RG=1460RG=1480RG=1500RG=1520RG=1540RG=1560RG=1580RG=1600RG=1620RG=1640RG=1660RG=1680RG=1700RG=1720RG=1740RG=1760RG=1780RG=1800RG=1820RG=1840RG=1860RG=1880RG=1900RG=1920RG=1940RG=1960RG=1980RG=2000RG=2020RG=2040RG=2060RG=2080RG=2100RG=2120RG=2140RG=2160RG=2180RG=2200RG=2220RG=2240RG=2260RG=2280RG=2300RG=2320RG=2340RG=2360RG=2380RG=2400RG=2420RG=2440RG=2460RG=2480RG=2500RG=2520RG=2540RG=2560RG=2580RG=2600RG=2620RG=2640RG=2660RG=2680RG=2700RG=2720RG=2740RG=2760RG=2780RG=2800RG=2820RG=2840RG=2860RG=2880RG=2900RG=2920RG=2940RG=2960RG=2980RG=3000RG=3020RG=3040RG=3060RG=3080RG=3100RG=3120RG=3140RG=3160RG=3180RG=3200RG=3220RG=3240RG=3260RG=3280RG=3300RG=3320RG=3340RG=3360RG=3380RG=3400RG=3420RG=3440RG=3460RG=3480RG=3500RG=3520RG=3540RG=3560RG=3580RG=3600RG=3620RG=3640RG=3660RG=3680RG=3700RG=3720RG=3740RG=3760RG=3780RG=3800RG=3820RG=3840RG=3860RG=3880RG=3900RG=3920RG=3940RG=3960RG=3980RG=4000RG=4020RG=4040RG=4060RG=4080RG=4100RG=4120RG=4140RG=4160RG=4180RG=4200RG=4220RG=4240RG=4260RG=4280RG=4300RG=4320RG=4340RG=4360RG=4380RG=4400RG=4420RG=4440RG=4460RG=4480RG=4500RG=4520RG=4540RG=4560RG=4580RG=4600RG=4620RG=4640RG=4660RG=4680RG=4700RG=4720RG=4740RG=4760RG=4780RG=4800RG=4820RG=4840RG=4860RG=4880RG=4900RG=4920RG=4940RG=4960RG=4980RG=5000RG=5020RG=5040RG=5060RG=5080RG=5100RG=5120RG=5140RG=5160RG=5180RG=5200RG=5220RG=5240RG=5260RG=5280RG=5300RG=5320RG=5340RG=5360RG=5380RG=5400RG=5420RG=5440RG=5460RG=5480RG=5500RG=5520RG=5540RG=5560RG=5580RG=5600RG=5620RG=5640RG=5660RG=5680RG=5700RG=5720RG=5740RG=5760RG=5780RG=5800RG=5820RG=5840RG=5860RG=5880RG=5900RG=5920RG=5940RG=5960RG=5980RG=6000RG=6020RG=6040RG=6060RG=6080RG=6100RG=6120RG=6140RG=6160RG=6180RG=6200RG=6220RG=6240RG=6260RG=6280RG=6300RG=6320RG=6340RG=6360RG=6380RG=6400RG=6420RG=6440RG=6460RG=6480RG=6500RG=6520RG=6540RG=6560RG=6580RG=6600RG=6620RG=6640RG=6660RG=6680RG=6700RG=6720RG=6740RG=6760RG=6780RG=6800RG=6820RG=6840RG=6860RG=6880RG=6900RG=6920RG=6940RG=6960RG=6980RG=7000RG=7020RG=7040RG=7060RG=7080RG=7100RG=7120RG=7140RG=7160RG=7180RG=7200RG=7220RG=7240RG=7260RG=7280RG=7300RG=7320RG=7340RG=7360RG=7380RG=7400RG=7420RG=7440RG=7460RG=7480RG=7500RG=7520RG=7540RG=7560RG=7580RG=7600RG=7620RG=7640RG=7660RG=7680RG=7700RG=7720RG=7740RG=7760RG=7780RG=7800RG=7820RG=7840RG=7860RG=7880RG=7900RG=7920RG=7940RG=7960RG=7980RG=8000RG=8020RG=8040RG=8060RG=8080RG=8100RG=8120RG=8140RG=8160RG=8180RG=8200RG=8220RG=8240RG=8260RG=8280RG=8300RG=8320RG=8340RG=8360RG=8380RG=8400RG=8420RG=8440RG=8460RG=8480RG=8500RG=8520RG=8540RG=8560RG=8580RG=8600RG=8620RG=8640RG=8660RG=8680RG=8700RG=8720RG=8740RG=8760RG=8780RG=8800RG=8820RG=8840RG=8860RG=8880RG=8900RG=8920RG=8940RG=8960RG=8980RG=9000RG=9020RG=9040RG=9060RG=9080RG=9100RG=9120RG=9140RG=9160RG=9180RG=9200RG=9220RG=9240RG=9260RG=9280RG=9300RG=9320RG=9340RG=9360RG=9380RG=9400RG=9420RG=9440RG=9460RG=9480RG=9500RG=9520RG=9540RG=9560RG=9580RG=9600RG=9620RG=9640RG=9660RG=9680RG=9700RG=9720RG=9740RG=9760RG=9780RG=9800RG=9820RG=9840RG=9860RG=9880RG=9900RG=9920RG=9940RG=9960RG=9980RG=10000
420 IF B>E+1 THEN YD=I-D:FORG=E+1 TO B-1:GOSUB375:YD=YD+I:NEXT:YD=YD+A
425 GOTO240

```



Electric Marathon



Har du en Marathon, Power eller Lambda kan dette program lave en række elektroniske beregninger. Blandt andet kan du indtaste bare to af følgende oplysninger: Spænding, strøm, modstand og effekt, hvorefter programmet foretager udregningen af de to manglende.

Hvis linierne fra 10 til 23, fra 490 til 760, og linie 135 udelades, kan programmet nøjes med bare 2 K RAM. □

lb Secher

```

SEREM
EL-BEREGNINGER
TIL LAMBDA 8300
OG MARATHON 32K
AF LB SECHER
1983

REM
CLS
10 PRINT AT 0,6;"**EL-BEREGNINGER**"
15 PRINT ""
20 PRINT AT 2,3;"TAST 0 FOR DE
UBEKENDTE,"
21 PRINT ""
22 PRINT AT 4,2;"DER SKAL VÆRE
E 2 UBEKENDTE."
23 PRINT ""
25 PRINT AT 6,0;"INDTAST WATT"
30 INPUT P
40 PRINT AT 6,17;P;" WATT"
50 PRINT AT 8,0;"INDTAST VOLT"
60 INPUT U
70 PRINT AT 8,17;U;" VOLT"
80 PRINT AT 10,0;"INDTAST AMPE
RE"
90 INPUT I
100 PRINT AT 10,17;I;" AMP"
110 PRINT AT 12,0;"INDTAST OHM"
120 INPUT R
130 PRINT AT 12,17;R;" OHM"

```

```

135 PRINT ""
140 IF R=0 AND P=0 THEN GOTO 20
150 IF U=0 AND I=0 THEN GOTO 35
160 IF U=0 AND R=0 THEN GOTO 30
170 IF I=0 AND P=0 THEN GOTO 25
180 IF I=0 AND R=0 THEN GOTO 40
190 IF U=0 AND P=0 THEN GOTO 45
200 P=U*I
210 PRINT AT 6,17;P;" WATT"
220 R=U/I
230 PRINT AT 12,17;R;" OHM"
240 GOTO 490
250 I=U/R
260 PRINT AT 10,17;I;" AMP"
270 P=(U*U)/R
280 PRINT AT 6,17;P;" WATT"
290 GOTO 490
300 U=P/I
310 PRINT AT 8,17;U;" VOLT"
320 R=P/(I*I)
330 PRINT AT 12,17;R;" OHM"
340 GOTO 490
350 I=50R (P/R)
360 PRINT AT 10,17;I;" AMP"
370 U=50R (R*P)
380 PRINT AT 8,17;U;" VOLT"
390 GOTO 490
400 I=P/U
410 PRINT AT 10,17;I;" AMP"
420 R=(U*U)/P
430 PRINT AT 12,17;R;" OHM"
440 GOTO 490

```

```

450 U=R*I
460 PRINT AT 8,17;U;" VOLT"
470 P=R*(I*I)
480 PRINT AT 6,17;P;" WATT"
490 PRINT AT 16,2;"ØNSKES FLERE
BEREGNINGER";AT 16,9;"TAST A"
495 PRINT AT 20,2;"FOR PRISUDRE
GNING TAST P"
520 IF INKEY$="A" THEN GOTO 5
530 IF INKEY$="P" THEN GOTO 535
531 IF INKEY$<"A" AND INKEY$<"P" THEN GOTO 490
535 CLS
536 PRINT "EFFEKTFORBRUGET = ";
P;" WATT"
540 PRINT "INDTAST TIDEN-I MINU
TTER"
550 INPUT T
560 Z=T*60
570 PRINT "INDTAST PRIS/KWH"
580 INPUT PRIS
700 G=INT (((P/1000)*(Z/3600))*
*PRIS)*100)/100
701 PRINT
702 PRINT
710 PRINT T;" MINUTTERS DRIFT"
720 PRINT "MED EN KWH-PRIS PAA
";PRIS;" KR"
730 PRINT "KOSTER ";G;" KRØNER"
740 GOTO 490
750 SAVE "E"
755 RUN

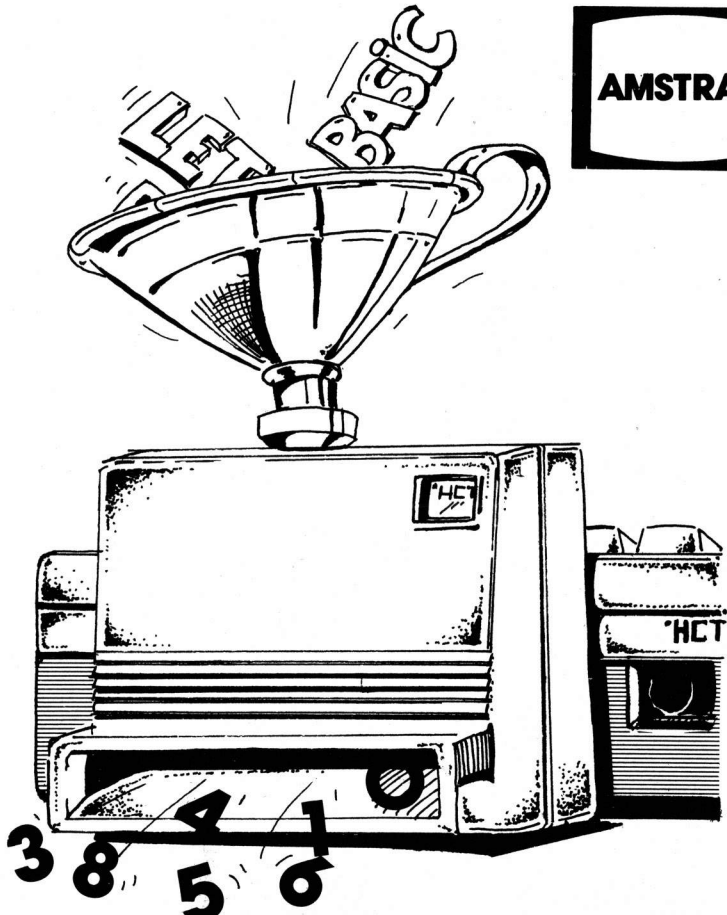
```

DISASSEM

```

10 *** (c)1984 Disassembler for AMSTRAD af Jens & Peter Ockelmann ***
20 *** INDSAET EVT. KONTROLKARAKTERER TIL PRINTER-INITIALISERING ***
100 *** INITIALISERING ***
110 MODE 1:BORDER 13:INK 0,13:INK 1,0:INK 3,3:PEN 1
120 DIM a$(16,2):FOR n=0 TO 16:FOR m=0 TO 7:READ a$(n,m):NEXT:NEXT
130 DATA B,C,D,E,H,L,QB,A
140 DATA BC,DE,QA,SP,?,?,?,?
150 DATA BC,DE,QA,AF,?,?,?,?
160 DATA 0,1,2,3,4,5,6,7
170 DATA NZ,Z,NC,C,PO,PE,P,M
180 DATA "ADD A","ADC A","SUB ","SBC A","AND ","XOR ","OR ","CP "
190 DATA NOP,"EX AF,AF","DJNZ QE,JE,?,?,?,?
200 DATA "(BC),A","A,(BC)","(DE),A","A,(DE)","(QD),QA","QA,(QD)","(QD),A","A,(QD)
)"
210 DATA RLCA,RRCA,RLA,RR,A,DAA,CPL,SCF,CCF
220 DATA RET,EXX,JP (QA),"LD SP,QA",?,?,?,?
230 DATA "IN A,(C)",?,"OUT (C),A","IN A,(C)",?,"EX (SP),QA","EX DE,HL",DI,EI
240 DATA 00,00,10,10,20,20,30,30
250 DATA RLC,RR,RL,RR,SLA,SRA,?,?"SRL "
260 DATA IM 0,?,IM 1,IM 2,?,?,?
270 DATA "LD I,A","LD R,A","LD A,I","LD A,R",RRD,RLD,?,?
280 DATA LD,CP,IN,OT,?,?,?,?
290 DATA ?,?,?,?,I,D,IR,DR
300 GOTO 1300
310 *** DISASSEMBLING AF KODEN I "byte" ***
320 f=(byte AND 192)/64:g=(byte AND 56)/8:h=byte AND 7:j=(g AND 6)/2:k=g AND 1
330 IF byte=118 THEN d$="HALT":RETURN
340 ON class+1 GOTO 350,360,370,380,390
350 ON +1 GOTO 360,370,380,390,370
360 ON +1 GOTO 410,430,450,460,480,490,500,510
370 ON +1 GOTO 540,550,570,580,590,600,620,630
380 ON +1 GOTO 640,650,660,670
390 IF <0 THEN 700
400 ON +1 GOTO 680,690,700,720,740,750,770,780
410 IF g>3 THEN d$="JR "+a$(4,g-4)+":":RETURN
420 d$=a$(6,g):RETURN
430 IF k=0 THEN d$="LD "+a$(1,j)+":":RETURN
440 d$="ADD QA","+a$(1,j):":RETURN
450 d$="LD "+a$(7,g):":RETURN
460 IF k=0 THEN d$="INC "+a$(1,j):":RETURN
470 d$="DEC "+a$(1,j):":RETURN
480 d$="INC "+a$(0,g):":RETURN
490 d$="DEC "+a$(0,g):":RETURN
500 d$="LD "+a$(0,g)+":":RETURN
510 d$=a$(0,g):RETURN
520 d$="LD "+a$(0,g)+":":RETURN
530 d$=a$(5,g)+a$(0,h):RETURN
540 d$="RET "+a$(4,g):":RETURN
550 IF k=0 THEN d$="POP "+a$(2,j):":RETURN
560 d$=a$(9,j):RETURN
570 d$="JP "+a$(4,g)+":":RETURN
580 d$=a$(10,g):RETURN
590 d$="CALL "+a$(4,g)+":":RETURN
600 IF k=0 THEN d$="PUSH "+a$(2,j):":RETURN
610 d$="CALL 00":":RETURN
620 d$=a$(5,g)+":":RETURN
630 d$="RST "+a$(11,g):":RETURN
640 d$=a$(12,g)+":":RETURN
650 d$="BIT "+a$(3,g)+":":RETURN
660 d$="RES "+a$(3,g)+":":RETURN
670 d$="SET "+a$(3,g)+":":RETURN
680 d$="IN "+a$(0,g)+":":RETURN
690 d$="OUT (C),":":RETURN
700 IF k=0 THEN d$="SBC HL","+a$(1,j):":RETURN
710 d$="ADC HL","+a$(1,j):":RETURN
720 IF k=0 THEN d$="LD (0D)","":RETURN
730 d$="LD "+a$(1,j)+":":RETURN
740 d$="NEG":":RETURN
750 IF k=0 THEN d$="RETN":":RETURN
760 d$="RETI":":RETURN
770 d$=a$(13,g):RETURN
780 d$=a$(14,g):RETURN
790 d$=a$(15,h)+a$(16,g):RETURN
800 STOP
810 *** INDSAET HL,IX,IY,(HL),(IX+d),(IY+d),N,NN HVIS PAAKRAEVET ***
820 I=INSTR(d$,"QA"):IF I<>0 THEN ON index+1 GOSUB 950,960,970
830 I=INSTR(d$,"QA"):IF I<>0 THEN ON index+1 GOSUB 950,960,970
840 I=INSTR(d$,"QB"):IF I<>0 THEN ON index+1 GOSUB 980,990,1010
850 I=INSTR(d$,"QC"):IF I<>0 THEN GOSUB 890:d$=LEFT$(d$,I-1)+q$+MID$(d$,I+2):a
a+1
860 I=INSTR(d$,"QD"):IF I<>0 THEN GOSUB 910:d$=LEFT$(d$,I-1)+q$+MID$(d$,I+2):a
a+2
870 I=INSTR(d$,"QE"):IF I<>0 THEN GOSUB 930:d$=LEFT$(d$,I-1)+q$+MID$(d$,I+2):a
a+1
880 RETURN
890 z=PEEK(a):GOSUB 1510:IF hf=1 THEN qc$=v$ ELSE qc$=c$
900 RETURN
910 z=PEEK(a)+256*PEEK(a+1):GOSUB 1500:IF hf=1 THEN qd$=v$ ELSE qd$=c$
920 RETURN
930 z=PEEK(a):GOSUB 1520:IF hf=1 THEN qe$=HEX$(x+1+a,4) ELSE qe$=STR$(x+1+a)
940 RETURN
950 d$=LEFT$(d$,I-1)+"HL"+MID$(d$,I+2):RETURN
960 d$=LEFT$(d$,I-1)+"IX"+MID$(d$,I+2):RETURN
970 d$=LEFT$(d$,I-1)+"IY"+MID$(d$,I+2):RETURN
980 d$=LEFT$(d$,I-1)+"HL"+MID$(d$,I+2):RETURN
990 z=PEEK(a):GOSUB 1530:IF hf=1 THEN c$=v$
1000 d$=LEFT$(d$,I-1)+"(IX+c$)+":MID$(d$,I+2):a=da+1:RETURN
1010 z=PEEK(da):GOSUB 1520:IF hf=1 THEN c$=v$
1020 d$=LEFT$(d$,I-1)+"(IY+c$)+":MID$(d$,I+2):a=da+1:RETURN
1030 *** HOVEDRUTINE ***
1040 ao=a
1050 IF ao>sluta THEN 1330
1060 a=a-1
1070 class=0:index=0
1080 da=a+3
1090 a=a+1
1100 byte=PEEK(a)
1110 IF class<0 THEN 1160
1120 IF byte=283 THEN class=1:GOTO 1090
1130 IF byte=237 THEN class=2:GOTO 1090
1140 IF byte=221 THEN index=1:GOTO 1090
1150 IF byte=253 THEN index=2:GOTO 1090
1160 IF class=1 AND index<0 THEN a=da+1:byte=PEEK(a)
1170 GOSUB 310
1180 a=a+1
1190 IF INSTR(d$,0)<>0 THEN GOSUB 810
1200 IF class=1 AND index<0 THEN a=da+1
1210 GOSUB 1240
1220 WHILE INKEY$<>:WEND:IF INKEY(47)=0 THEN GOSUB 1350
1230 GOTO 1030
1240 *** UDPRINTNING ***
1250 IF hf=1 THEN PRINT$TRM,HEX$(ao,4):":":FOR n=ao TO a-1:PRINT$TRM,HEX$(PE
EK(n),2):":":NEXT:PRINT$TRM,TAB(25)ID$
1260 IF hf=0 THEN PRINT$TRM,USING"#####":ao:PRINT$TRM,"":FOR n=ao TO a-1:PR
INT$TRM,STR$(PEEK(n)):NEXT:PRINT$TRM,TAB(25)ID$
1270 t=t+1:IF t>24 THEN WHILE INKEY$<>:WEND:PEN 3:PRINT "ICHR$(24):"ENTER"IC
HR$(24)":":FOR n=ao TO a-1:PRINT$TRM,HEX$(PEEK(n),2):":":NEXT:PRINT$TRM,TAB(25)ID$
1280 LOCATE 1,25:PEN 1
1290 RETURN
1290 GOTO 1030
1300 *** MENU 1 ***
1310 INPUT"dec / hex ? (d/h) ",i$
1320 hf=1+(LOWER$(i$)="d")
1330 GOSUB 1350

```



Eksempel: Et stykke af 'jumpblokken'

disassemb i hex.		...og samme stykke i decimal.	
B9D2: 06 DF	LD B,DF		
B9D4: ED 79	OUT (C),A	47578: 6 223	LD B, 223
B9D6: 21 AB B1	LD HL,B1AB	47572: 237 121	OUT (C),A
B9D9: 46	LD B,(HL)	47574: 33 168 177	LD HL, 45480
B9DA: 77	LD (HL),A	47577: 70	LD B,(HL)
B9DB: C5	PUSH BC	47578: 119	LD (HL),A
B9DC: FD E5	PUSH IY	47579: 197	PUSH BC
B9DE: 3D	DEC A	47580: 253 229	PUSH IY
B9DF: FE 07	CP 07	47582: 61	DEC A
B9E1: 38 0F	JR NC,B9F2	47583: 254 7	CP 7
B9E3: 07	ADD A,A	47585: 48 15	JR NC, 47602
B9E4: C6 AC	ADD A,AC	47587: 135	ADD A,A
B9E6: 6F	LD L,A	47588: 198 172	ADD A, 172
B9E7: CE B1	ADC A,B1	47589: 111	LD L,A
B9E9: 95	SUB L	47591: 286 177	ADC A, 177
B9EA: 67	LD H,A	47593: 149	SUB L
B9EB: 7E	LD A,(HL)	47594: 103	LD H,A
B9EC: 25	INC HL	47595: 126	LD A,(HL)
		47596: 35	INC HL

```

1340 GOTO 1030
1350 *** MENU 2 ***
1360 PRINT"startadresse ";IF hf=0 THEN PRINT"(dec) "; ELSE PRINT"(hex) ";
1370 INPUT"adr$":IF adr$="" THEN 1360
1380 PRINT"slutadresse ";IF hf=0 THEN PRINT"(dec) "; ELSE PRINT"(hex) ";
1390 INPUT"slutadr$":IF slutadr$="" THEN slutadr$="0"
1400 IF adr$="" THEN 1360
1410 INPUT"Ønskes tekst paa printer (last P) ",i$
1420 IF LOWER$(i$)<>"p" THEN strm=0:GOTO 1450
1430 strm=i$
1440 PRINT"hold [SPACE] nede for at afbryde"
1450 IF hf=0 THEN RESTORE 1450:FOR n=0 TO 7:READ a$(11,n):NEXT:a=VAL(adr$) ELSE
a=VAL("h"+adr$:DATA 00,00,16,24,32,40,48,56
1460 IF hf=0 THEN sluta=VAL(slutadr$) ELSE sluta=VAL("h"+slutadr$)
1470 IF sluta<0 THEN sluta=65536+sluta
1480 IF a<0 THEN a=65536+a
1490 t=t+3:RETURN
1500 c$=STR$(z):v$=HEX$(z,4):RETURN
1510 c$=STR$(z):v$=HEX$(z,2):RETURN
1520 x=z:IF x>127 THEN x=x-256
1530 IF x<0 THEN c$="--"+STR$(ABS(x)) ELSE c$="++"+STR$(x)
1540 IF x<0 THEN v$="--"+HEX$(ABS(x),2) ELSE v$="++"+HEX$(x,2)
1550 RETURN

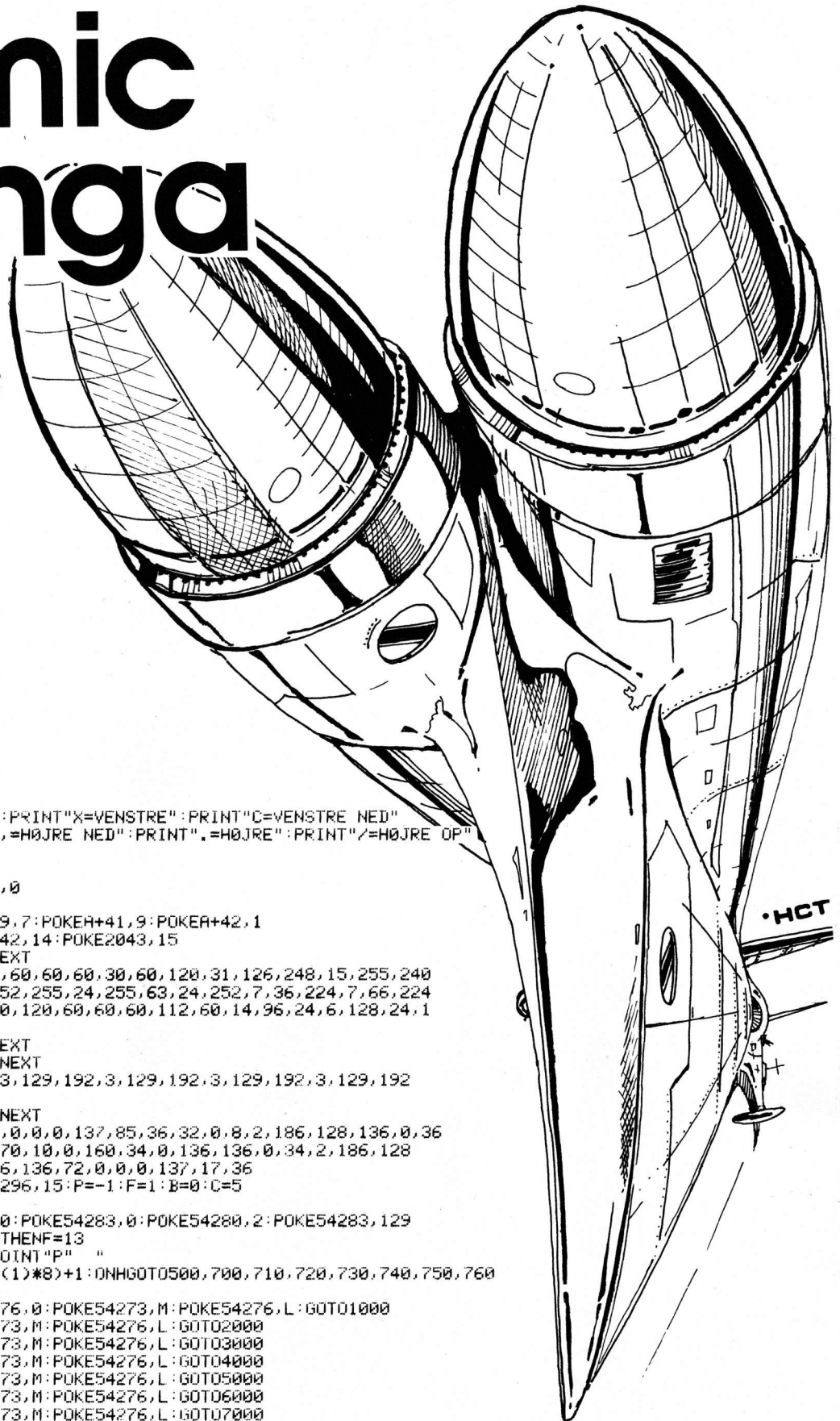
```


Cosmic Changa

Du befinder dig i en firkantet rumstation, som kan skyde i 8 retninger.

Du bliver angrebet af UFO'er fra kampstjernen Changa. De skal selvfølgelig skydes ned, inden de når din rumstation. Rammes stationen af en UFO, sprænges den i luften, og spillet er slut. »Cosmic Changa« er selvfølgelig og bliver sværere med tiden.

Brian Rasmussen

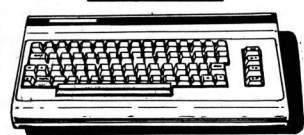


```

0 PRINT"Q"
1 FORK=54272T054296:POKEK,0
2 M=255:L=129
3 POKE54285,5
4 POKE54277,10:POKE54284,11
5 PRINT"KEYS":PRINT"Z=VENSTRE OP":PRINT"X=VENSTRE":PRINT"C=VENSTRE NED"
6 PRINT"V=NED":PRINT"M=OP":PRINT",=HØJRE NED":PRINT".=HØJRE":PRINT"/=HØJRE OP"
7 PRINT"ANY KEY TO CONTINUE"
8 IFPEEK(197)=64THEN8
10 PRINT"Q":POKE53280,0:POKE53281,0
20 A=53248
30 POKER+21,15:POKER+40,1:POKER+39,7:POKER+41,9:POKER+42,1
40 POKE2040,11:POKE2041,13:POKE2042,14:POKE2043,15
50 FORK=0T062:READB:POKE704+K,B:NEXT
60 DATA128,24,1,96,24,6,112,60,14,60,60,60,30,60,120,31,126,248,15,255,240
70 DATA7,129,224,7,66,224,63,36,252,255,24,255,63,24,252,7,36,224,7,66,224
80 DATA15,255,240,31,126,248,30,60,120,60,60,60,112,60,14,96,24,6,128,24,1
90 POKER,128:POKER+1,128
100 FORK=0T04STEP3:POKEK+832,24:NEXT
110 FORK=0T028:READB:POKE896+K,B:NEXT
120 DATA0,60,0,0,255,0,1,195,128,3,129,192,3,129,192,3,129,192,3,129,192
130 DATA1,195,128,0,255,0,0,60
140 FORK=0T062:READB:POKEK+960,B:NEXT
150 DATA137,17,34,0,0,0,36,146,72,0,0,0,137,85,36,32,0,0,2,186,128,136,0,36
160 DATA36,0,136,10,0,160,170,0,170,10,0,160,34,0,136,136,0,34,2,186,128
170 DATA32,0,8,137,185,66,0,0,0,36,136,72,0,0,0,137,17,36
200 POKER+39,7:POKE54278,1:POKE54296,15:P=-1:F=1:B=0:C=5
300 POKER+6,B:POKER+7,C=5
301 POKER+41,RND(1)*15+1:POKER+2,0:POKE54283,0:POKE54280,2:POKE54283,129
302 IFP=0THEN0=0+10:F=F+.5:IFF>13THENF=13
305 P=P+1:PRINT"Ø":PRINTSPC(31)"POINT"P"
310 POKER+7,0:POKER+6,0:H=INT(RND(1)*8)+1:ONHGOTO500,700,710,720,730,740,750,760
500 B=0:C=6:Z=F:X=F
510 Q=PEEK(197):IFQ=12THENPOKE54276,0:POKE54273,M:POKE54276,L:GOTO1000
520 IFQ=23THENPOKE54276,0:POKE54273,M:POKE54276,L:GOTO2000
530 IFQ=20THENPOKE54276,0:POKE54273,M:POKE54276,L:GOTO3000
540 IFQ=31THENPOKE54276,0:POKE54273,M:POKE54276,L:GOTO4000
550 IFQ=36THENPOKE54276,0:POKE54273,M:POKE54276,L:GOTO5000
560 IFQ=47THENPOKE54276,0:POKE54273,M:POKE54276,L:GOTO6000
570 IFQ=44THENPOKE54276,0:POKE54273,M:POKE54276,L:GOTO7000
580 IFQ=55THENPOKE54276,0:POKE54273,M:POKE54276,L:GOTO8000
590 B=B+X:C=C+Z:POKER+4,B:POKER+5,C
600 IFC+5>128ANDC-15<128ANDB+10>128ANDB-10<128THEN9000
610 GOT0510
700 C=133:B=0:X=F:Z=0:GOTO510
710 C=255:B=5:X=F:Z=-F:GOTO510
720 C=255:B=128:Z=-F:X=0:GOTO510
730 C=255:B=250:X=-F:Z=-F:GOTO510
740 C=133:B=255:X=-F:Z=0:GOTO510
750 C=8:B=255:X=-F:Z=F:GOTO510
760 B=128:C=0:X=0:Z=F:GOTO510
1000 D=136:FORK=130T040STEP-8:D=D-8:POKER+2,D:POKER+3,K:B=B+X:C=C+Z:POKER+4,B

```

COMMO-
DORE
64



Snyd (ikke)

S 13

Dem selv i skat!

■ Dette skatteprogram beregner en række faktorer ud fra en række indtastede data. For at programmet skal virke fuldt ud, skal flg. data indtastes:

1. Kommuneskat %
2. Kirkeskat %
3. Amtsskat %
4. Personfradrag
5. 1. Skattegrænse i %
6. 2. Skattegrænse i %
7. 3. Skattegrænse i %
8. 4. Skattegrænse i kr.
9. 2. Skattegrænse i kr.
10. Samlet pensionsfonds %
11. Statsskatte %

12. Skattepligtig indkomst

13. A-indkomst

14. Betalt skat

Flg. kan da beregnes:

1. Trækprocent
2. Skattekortsfra drag
3. Skat i kr.
4. Skat i % af skattepligtig indkomst
5. Skat i % af A-indkomst
6. Restskat i kr.
7. Restskat i % af korrekt skat.

Udover det er programmet selvinstruerende. □

Carsten Andersen

```
10 REM *****
11 REM *****SKAERMBILLEDE*****
12 REM *****
13 REM
15 PRINT "J":POKE53280,6:POKE53281,6:PRINT "J"
20 GOSUB23000
1000 REM
1001 REM *****
1002 REM *INDTASTNING AF SKATTEDATA*
1003 REM *****
1004 REM
1050 PRINT "J":PRINT "*****SKATTEPROGRAM 1"
1060 PRINT "*****AF CARSTEN ANDERSEN 1984."
1070 PRINT "*****"
1080 INPUT "001. INDTAST KOMMUNESKAT% ";KS
1100 INPUT "002. INDTAST KIRKESKAT% ";KS
1200 INPUT "003. INDTAST AMTSSKAT% ";AS
1300 INPUT "004. INDTAST PERSONFRADRAG ";PF
1400 INPUT "005. INDTAST 1. SKATTEGRAENSE (%);SG
1500 INPUT "005. INDTAST 2. SKATTEGRAENSE (%);SL
1600 INPUT "007. INDTAST 3. SKATTEGRAENSE (%);SF
1700 INPUT "009. INDTAST 1. SKATTEGRAENSE (KR);S1
1800 INPUT "009. INDTAST 2. SKATTEGRAENSE (KR);S2
1900 INPUT "010. INDTAST SAMLET PENSIONSFOND% ";PS
1930 INPUT "011. INDTAST STATSSKAT% ";S
2000 INPUT "012. INDTAST SKATTEPLIGTIG INDKOMST ";SI
2010 INPUT "013. INDTAST A-INDKOMST ";AI
2020 INPUT "014. INDTAST BETALT SKAT ";BS
4000 REM
4001 REM *****
4002 REM *****BEREGNINGERNE*****
4003 REM *****
4004 REM
4005 PRINT "J":POKE53280,2:POKE53281,2:PRINT "J"
4006 T=KS+CS+AS
4008 PRINT "*****BEREGNINGERNE"
4009 PRINT "*****"
4010 PRINT "001. TRAEKPROCENT."
4020 PRINT "002. SKATTEKORTSFRA DRAG."
4030 PRINT "003. SKAT I KR."
4040 PRINT "004. SKAT I % AF SKATPLG. INDK."
4050 PRINT "005. SKAT I % AF A-INDKOMST."
4060 PRINT "006. RESTSKAT I KR."
4070 PRINT "007. RESTSKAT I % AF KORREKT SKAT."
4071 PRINT "008. EDITERING AF DATA"
4075 PRINT "009. INDTAST TALLE T UD FOR BEREGNINGEN."
4080 GETB$:IFB$="" THEN4080
4090 IFB$="1" THENGOSUB5000
4091 IFB$="2" THENGOSUB6000
4092 IFB$="3" THENGOSUB7000
4093 IFB$="4" THENGOSUB8000
4094 IFB$="5" THENGOSUB9000
4095 IFB$="6" THENGOSUB10000
4096 IFB$="7" THENGOSUB11000
4097 IFB$="8" THENGOSUB13000
4098 RETURN
5000 REM
```



```
PERSONNUMMER
5001 REM *****
5002 REM *BEREGNING AF TRAEKPROCENTEN*
5003 REM *****
5004 REM
5006 IFSK=0THENGOSUB7000
5010 T1=(PF*(SG+T+PS)/100)+SK
5020 TP=(T1*100)/SI
5030 IFAI>S1 THENTP=INT(TP+1.5)
5040 IFAI<=S1 THENTP=INT(TP+1)
5045 IFB$="2" THENRETURN
5047 PRINT "J":POKE53280,2:POKE53281,2:PRINT "J"
5050 PRINT "*****ANDERES TRAEKPROCENT = ";TP;"%"
5051 PRINT "*****ANDERES SKATTEKORTSFRA DRAG = ";SD;"KR."
5052 GETA$: IFA$="" THEN5052
5053 IFA$="J" THEN4000
5054 IFA$="N" THENRUN
5055 IFA$<>"J" OR A$<>"N" THEN5052
5060 RETURN
6000 REM
6001 REM *****
6002 REM *BEREGN. AF SKAT.KORTFRADRAG*
6003 REM *****
6004 REM
6020 IFSK=0THENGOSUB7000
6025 IFTP=0THENGOSUB5000
6030 SD=INT((AI-(SK*100/(INT(TP*10)/10)))/12)
6035 PRINT "J":POKE53280,6:POKE53281,6:PRINT "J"
6040 PRINT "*****ANDERES SKATTEKORTSFRA DRAG = ";SD;"KR."
6051 PRINT "*****ANDERES SKATTEKORTSFRA DRAG = ";SD;"KR."
6052 GETA$: IFA$="" THEN6052
6053 IFA$="J" THEN4000
6054 IFA$="N" THENRUN
6055 IFA$<>"J" OR A$<>"N" THEN6052
6060 RETURN
7000 REM
7001 REM *****
7002 REM *****BEREGN. AF SKAT*****
7003 REM *****
7004 REM
7005 IFSI<=S1 THENGOSUB12000
7010 IFSI>S1 AND SI<=S2 THENGOSUB12400
7020 IFSI>S2 THENGOSUB12800
7025 S9=(SI/100)*PS
7030 S0=(SI/100)*T
7035 S0=(PF/100)*(T+S+PS)
7040 SK=S5+S9+S0-S0
7045 IFB$="1" OR B$="2" OR B$="4" OR B$="5" OR B$="6" OR B$="7" THENRETURN
7085 PRINT "J":POKE53280,2:POKE53281,2:PRINT "J"
7090 PRINT "*****DE SKAL BETALE ";SK;" KR I SKAT."
7091 PRINT "*****"
7092 GETA$: IFA$="" THEN7092
7093 IFA$="J" THEN4000
7094 IFA$="N" THENRUN
7095 IFA$<>"J" OR A$<>"N" THEN7092
7096 RETURN
8000 REM
8001 REM *****
8002 REM *SKAT I % AF SKATPLG. INDK.*
8003 REM *****
8004 REM
8010 PRINT "J":POKE53280,2:POKE53281,2:PRINT "J"
8020 IFSK=0THENGOSUB7000
8030 SA=INT((SK*100)/SI)
8040 PRINT "*****ANDERES SKAT I % AF SKATPLG. INDK. ";SA;" %"
8042 PRINT "*****ANDERES SKAT I % AF A-INDK. ";SN;" %"
8043 GETA$: IFA$="" THEN8043
8044 IFA$="J" THEN4000
8045 IFA$="N" THENRUN
8046 IFA$<>"J" OR A$<>"N" THEN8043
8050 RETURN
9000 REM
9001 REM *****
9002 REM **SKAT I % AF A-INDK.**
9003 REM *****
9004 REM
9020 IFSK=0THENGOSUB7000
9030 SN=INT((SK*100)/AI)
9035 PRINT "J":POKE53280,2:POKE53281,2:PRINT "J"
9040 PRINT "*****ANDERES SKAT I % AF A-INDK. ";SN;" %"
9042 PRINT "*****ANDERES SKAT I % AF A-INDK. ";SN;" %"
9043 GETA$: IFA$="" THEN9043
9044 IFA$="J" THEN4000
```


HVEM GI'R DIG FREM



FREMTIDENS SOFTWARE?



Commodore!

Billedet viser bare et lille udpluk af programmer til Commodore 64, verdens mest populære hjemme-computer.

Commodore er faktisk det computerfirma i verden, der har udviklet flest programmer overhovedet, ca. 2.000 ialt - foreløbig!

I realiteten findes der et program til enhver opgave, du kan tænke dig - plus en hel del andre. Og der kommer stadig nye til - der står Commodore på fremtidens software.



Commodore

Fordi fremtiden forlængst er begyndt.

Verdens sjoveste computer-kursus

Alt om
DATA
Klubben

I samarbejde med Cavendish i London præsenterer "Alt om Data" markedets sjoveste og mest farverige computerkursus. Det består af ugebladet INPUT, der på letlæst engelsk giver dig alle de tips og tricks, du har brug for, når du arbejder med de mest populære hjemmedatamater. "Alt om Data Klubben" sender dig hver måned fire fortløbende og trykfriske eksemplarer af INPUT. Hver gang får du et væld af usædvanligt flotte programmer og listnin-

ger. Plus en masse gode råd og tips for både begynderen og den mere erfarne hjemmecomputer ejer. Det er ikke dyrt at være medlem af "Alt om Data Klubben". Hvert INPUT magasin koster kr. 18,75, og du bestiller 4 ad gangen – altså en måneds udsendelse for kr. 75. Langt mindre end en investering i et købeprogram. Ud over INPUT kommer "Alt om Data Klubben" løbende med spændende konkur-

rencer og særtilbud til medlemmerne. INPUT computerkurserne henvender sig til alle, som vil lære mere om data på en forhøjelig måde. De viste programlistninger passer til de fleste computere, – nemlig Commodore 64, Spectrum, VIC-20, ZX-81, Dragon, BBC; Electron og Colour Genie. Bliv medlem af "Alt om Data Klubben". Brug evt. girokortet i "Alt om Data" nummer 8 eller rekvirer et nyt på tlf.: 01-11 28 33. Eller send os blot 75 kr. + navn og adresse.



"Alt om Data Klubben" er stiftet af månedsbladet "Alt om Data", der er Danmarks største computermagasin. Her kan du hver måned finde spændende artikler fra hele verden, grundige test-rapporter og masser af programmer.