

## Seymour Papert i Danmark: »Datalære som selvstændigt fag er en misforståelse«

En af kapaciteterne inden for området EDB og undervisning - Seymour Papert - besøgte Danmark i dagene 14-15. maj. Seymour Papert er ansat ved det kendte MIT-universitet i Boston, og er især kendt for at have udviklet programmeringssproget LOGO, der bl.a. indeholder den »Turtle-Graphics«, som benyttes i de to sprog »MIKROLOGO« og »MYRESNAK«. Det skal dog med det samme slås fast, at LOGO er et sprog, der går langt ud over denne meget begrænsede del. LOGO kan benyttes ligesåvel som andre programmeringssprog til almindelig talbehandling, tekstbehandling, filopbygning

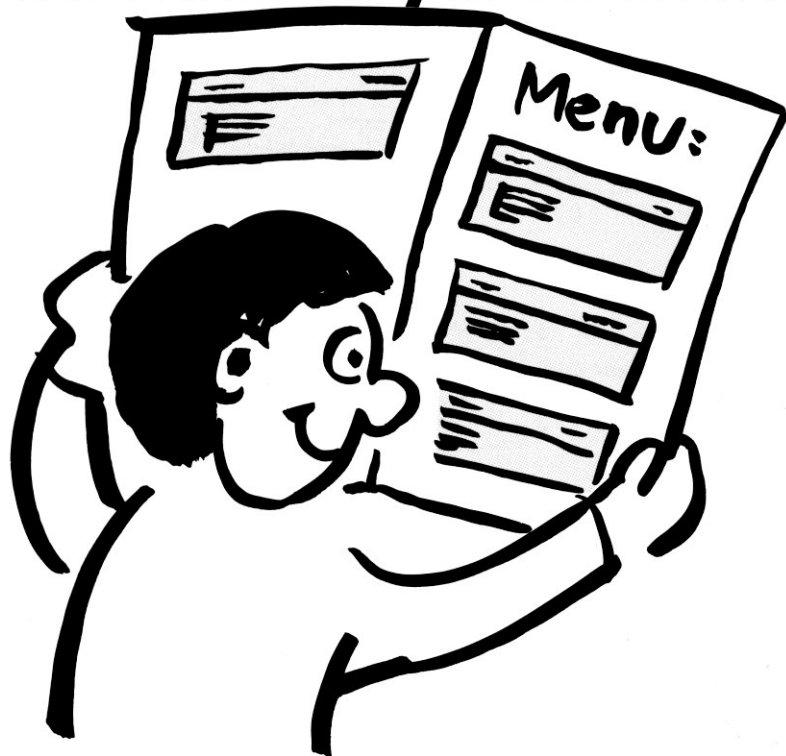
osv. Der findes dog ikke nogen fuldstændige, danske varianter af LOGO.

Seymour Papert, der var inviteret af Dansk Center for Pædagogik og Informatik, talte d. 14. ved en konference på Christiansborg, hvor der var ca. 250 tilhørere fra uddannelsessektoren, pressen og forskellige EDB-leverandører.

Selvom Seymour Papert kun var i landet to dage, lykkedes det alligevel PICCOLINIEN at få ham til at bruge en aften på et interview, især fordi det netop er det undervisningsmæssige aspekt, der interesserer ham mest.

Fortsættes på midtersiderne.

## Anvendelse af menu-systemet



I sidste nummer af PICCOLINIEN omtalte vi det indbyggede menu-system, der kunne benyttes i forbindelse med filflytninger, konfigureringer o.lign.

I dette nummer skal vi se på, hvordan man selv kan opbygge menuer, der sætter elever i stand til at vælge et bestemt undervisningsprogram, ved et tryk på en enkelt knap. Man opnår herved, at man i en ren programanvendelsesfase ikke behøver tænke på ordrer som LOAD, RUN o.lign.

Man opbygger sine egne menuer ved hjælp af et program - MENUVEDL - hvis navn er en forkortelse for menuvedligeholdelse, og som både benyttes ved opbygning af nye menuer og ændring af gamle menuer.

Programmet kaldes fra det yderste CCP/M-niveau ved at skrive MENUVEDL <retur>. Derefter er programmet i høj grad selvforklarende, idet det selv er menu-opbygget.

Vi skal her prøve at forklare anvendelsen af MENUVEDL ved at gennemgå et konkret eksempel.

Vi forestiller os, at vi vil lave en menu, hvor eleverne kan vælge mellem følgende programmer:

1. MYRESNAK  
- der er et tidligere omtalt tegneprogram.
2. ANNUITET  
- et Comal80-program til beregning af ydelser på annuitetslån. Det benytter hverken grafik eller datafiler
3. KOST  
- Kostanalyse: et Comal80-program, der hele tiden læser i en datafil.
4. PERSPEKT  
- Comal80-program, der simulerer perspektivtegning. Benytter grafik.
5. RCTEKST  
- Tekstbehandlingssystem, der startes gennem en anden menu.

Efter starten af MENUVEDL viser der sig på skærmen, den menu der er afbildet i fig. 1.

Da det er en ny menu vi skal lave, har vi ikke brug for at hente tidligere definerede menuer, men starter med at tilføje en menulinie i en tom menu. Vi trykker derfor T, og bliver derefter bedt om at angive følgende tre ting:

1. VALGTEGN  
- dvs. det bogstav eleven skal taste for at starte det ønskede program.
2. TEKST  
- dvs. den forklarende tekst, som eleven ser. Denne tekst er dog begrænset til 45 tegn på en linie, hvilket dog kan ændres af brugeren.
3. KOMMANDOLINIE  
- dvs. de(n) ordre(r), der skal udføres for at programmet starter og efter afslutning vender tilbage til menuen. Denne linie ser eleven *ikke* i menubilledet.

Det er som oftest punkt 3, der er det vanskelige. Vi skal starte med menulinien hørende til myresnak, og svarer ved de to første punkter:

1. M (kan vælges frit, men ofte benyttes første bogstav i teksten).

2. MYRESNAK - et tegneprogram.

Ved punkt 3 skal man huske, at MYRESNAK kræver, at grafiksyste- met er installeret, og at det endvidere vil være en fordel at få dette fjernet efter udførelsen af MYRESNAK, for at det ikke skal fylde i lageret. Vi skriver derfor:

```
3. GRAPHICS//MYRESNAK//  
   GRAPHICS NO
```

hvor de to sæt dobbeltstreger bruges til at adskille flere ordrer på samme linie (NB: Fungerer kun på CCP/M 3.1 systemer).

Vi har nu et billede som vist i fig. 2, hvor systemet spørger, om man ønsker menu-linien tilføjet (fortrydelsesmulighed). Svarer vi j på dette sted, går vi i gang med næste menulinie.

I fig. 3-6 har vi vist opbygningen for de fire andre menuliner. Vi skal her kun kommentere punkt 3 i hver af dem.

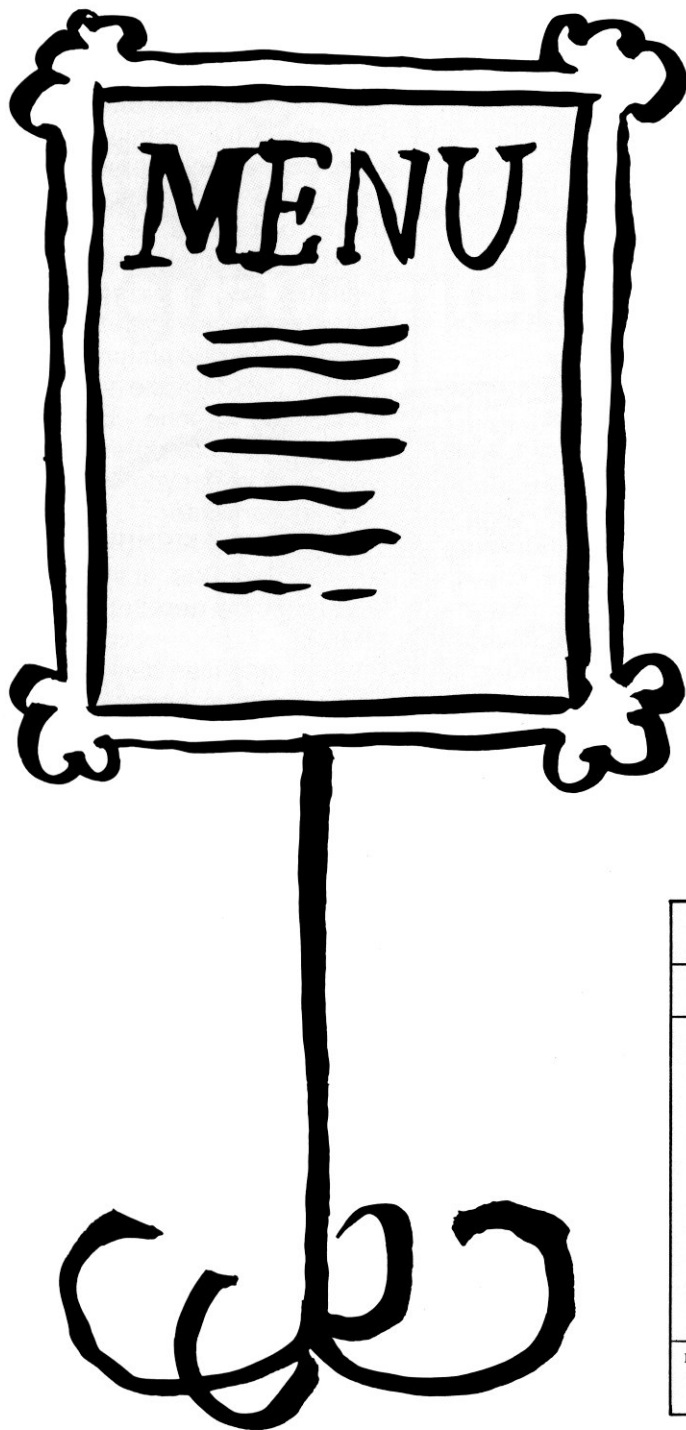
Fig. 3.

Annuitetsprogrammet benytter ikke datafiler eller grafik. Man skal blot starte Comal80 med programnavnet lige bagefter, så starter udførelsen af ANNUITET, når Comal80 er startet (se evt. PICCOLINIEN 85.2).

Fig. 4.

KOST-programmet benytter hele tiden en datafil kaldet DATA. Arbejder man i en 4-brugersløsning, er det upraktisk, hvis man fra alle 4 maskiner hele tiden skal ned og læse på den samme diskette. Man må derfor sørge for at flytte datafilen over M-disk'en, således at hver enkelt maskine fungerer lokalt. Endvidere

# PICCOLINIEN



bliver tilgangstiden til filen væsentlig mindre på denne måde. Overførslen sker ved hjælp af PIP-ordren.

Derefter sørger vi for at gøre M-disk'en til vores aktive disk, for at vi læser i datafilen det rigtige sted, ved at skrive M:, og så startes Comal80 fra A-disk'en og med KOST fra A-disk'en til automatisk opstart.

Efter afslutningen af Comal80, returnerer vi til A-disk'en og sletter (ERA) filen DATA fra M-disk'en, så den ikke ligger og fylder op. Den originale ligger selvfølgelig stadig på A-disk'en. Fig. 5.

Perspektiv-programmet benytter grafik, men ikke datafiler. Det er derfor ikke nødvendigt at benytte M-disk'en. Vi skal blot sørge for, at grafikken bliver installeret inden opstarten af Comal80, og fjernet igen efter afslutningen. Fig. 6.

På de nyeste udgaver af tekstbehandlingssystemet RcTekst ligger der et menu-system, der benyttes ved opstart. Dette system sikrer, at man i

Fig. 2

PICCOLINE Menuvedligeholdelse		Version 2.1
Valgtegn	M	
Tekst	MYRESNAK -et tegneprogram	
Kommandolinie	GRAPHICS//MYRESNAK//GRAPHICS NO	
OK at tilføje menulinie ? (j/n)		
Tryk ESC for at returnere		
Konsol=0 Dynamisk MENUVEDL		Skriver=0 LÅS 13:25:20

Fig. 3

PICCOLINE Menuvedligeholdelse		Version 2.1
Valgtegn	A	
Tekst	BEREGNING AF YDELSER PÅ ANNUITETSLÅN	
Kommandolinie	COMAL80 ANNUITET	
OK at tilføje menulinie ? (j/n)		
Tryk ESC for at returnere		
Konsol=0 Dynamisk MENUVEDL		Skriver=0 LÅS 13:36:33

Fig. 1

PICCOLINE Menuvedligeholdelse		Version 2.1
H	Hent menudefinition	
G	Gem menu	
I	Indhold af menu	
R	Rediger menulinie	
T	Tilføj menulinie	
S	Slet menulinie	
F	Skærmformat for menu	
O	Re-organisér menu	
Tryk ESC for at returnere		
Konsol=0 Dynamisk MENUVEDL	Skriver=0	13:23:57

# PICCOLINIEN

en 4-brugerløsning ikke får blokeret disktestationen bl.a. ved endnu en gang at benytte M-disk'en. I vores menu vil vi blot sørge for, at vi kommer over til denne nye menu. Denne menu har fået navnet TEKST, og da vi allerede er inde i menu-systemet, skriver vi blot dette navn efterfulgt af bogstaverne MDF, der er en forkortelse for menu-definition.

Efter indskrivningen af den sidste linie bliver man bedt om det næste valgtegn. Da man ikke ønsker flere, trykker man ESC og returnerer nu til hovedmenuen. Man kan nu se indholdet af sin menu ved at trykke I. Gør vi det, vil vi se, at vi i vores menu mangler overskrifter o.lign. Vi vender tilbage til hovedmenuen ved at trykke ESC, og trykker F for Format. Det bringer os over til en ny menu, der bl.a. indeholder en linie 8 til brug for overskriftlinie. Denne linie 8 er ikke umiddelbart synlig, da der kun er plads til 7 linier i menuen. Men ved et tryk - som angivet i nederste linie - på knappen A4, kommer man frem til de

næste linier. Dette princip fungerer automatisk ved alle menuer - også egne - der ikke kan være i et skærm-billede.

Vi indtaster en passende tekst (max. 78 tegn) og trykker retur. Derefter vælges l for at gemme dette menuformat, hvorefter vi igen returnerer til hovedmenuen ved at trykke ESC. Trykker vi nu I, kan vi se vores menu, der er vist i fig. 7.

Man kan fra hovedmenuen foretage en række forskellige redigeringer i sin menu, hvis man ønsker det. Efter færdigredigering skal menuen gemmes, ved at man trykker G. Man bliver så bedt om et navn, og vi kune vælge at kalde denne menu for ELEV.

Når vi har gemt menuen afsluttes MENUVEDL ved at trykke ESC, og vi kan nu kalde menuen ved at skrive MENU ELEV.

Vi skal til sidst se på, hvordan denne diskette kan gøres til et »lukket« system.

For det første skal vi sørge for, at vi, hver gang vi har afsluttet et Comal80-

program, også får afsluttet Comal80 således at vi vender tilbage til menuen, vi kaldte fra. Man returnerer fra Comal80 v.h.a. kommandoen BYE, men denne fungerer også som en ordre inde i et program. Man skal derfor blot sørge for, at den sidste linie i programmet bliver ordren BYE. Desuden skal vi sikre os, at vi ved start af maskinen kommer direkte ind i menuen. I forrige nummer af PICCOLINIEN (feb.85) viste vi et lille program, der kunne benyttes ved ændring af opstartfilen STARTUP.0. Dette kan vi benytte til at lave følgende opstartslinie:

PRINTER 2//MENU ELEV  
der først bevirker, at vi får en fælles printer på, og derefter starter vores menu.

Til sidst skal man huske, at nogle programmerne benytter M-disk, således at man skal sørge for, at maskinen bliver konfigureret med en sådan.

Menuen forudsætter, at disketten placeres i disktestation A.

Fig. 4

PICCOLINE Menuvedligeholdelse		Version 2.1
Valgtegn	K	
Tekst	KOST - VITAMINER/MINERALER I MADVARER	
Kommandolinie	PIP M:=A:DATA//M://A:COMAL80 A:KOST//A://ERA M:DATA	
OK at tilføje menulinie ? (j/n)		
Tryk ESC for at returnere		
Konsol=0 Dynamisk MENUVEDL	Skriver=0	LÅS 13:38:35

Fig. 6

PICCOLINE Menuvedligeholdelse		Version 2.1
Valgtegn	T	
Tekst	TEKSTBEHANDLINGSSYSTEMET RCTEKST	
Kommandolinie	TEKST.MDF	
OK at tilføje menulinie ? (j/n)		
Tryk ESC for at returnere		
Konsol=0 Dynamisk MENUVEDL	Skriver=0	LÅS 15:36:01

Fig. 5

PICCOLINE Menuvedligeholdelse		Version 2.1
Valgtegn	P	
Tekst	PERSPEKTIV-TEGNINGS-PROGRAM	
Kommandolinie	GRAPHICS//COMAL80 PERSPEKT//GRAPHICS NO	
OK at tilføje menulinie ? (j/n)		
Tryk ESC for at returnere		
Konsol=0 Dynamisk MENUVEDL	Skriver=0	LÅS 14:02:30

Fig. 7

VÆLG ET AF NEDENSTÅENDE PROGRAMMER VED AT TASTE VALGTEGNET		
M	MYRESNAK -et tegneprogram	
A	BEREGNING AF YDELSER PÅ ANNUITETSLÅN	
K	KOST - VITAMINER/MINERALER I MADVARER	
P	PERSPEKTIV-TEGNINGS-PROGRAM	
T	TEKSTBEHANDLINGSSYSTEMET RCTEKST	
Tryk ESC for at returnere		
Konsol=0 Dynamisk MENUVEDL	Skriver=0	LÅS 13:42:34

## Katalog over EDB-litteratur og undervisningsprogrammel



I sidste nummer af PICCOLINIEN omtalte vi meget kort det nye katalog, som vi her fra redaktionen vil udsende.

Det drejer sig om et katalog over undervisningsprogrammel, der kan benyttes på PICCOLINIEN, samt søger til brug i forbindelse med EDB-undervisning.

Udsendelsen af kataloget skal ses i lyset af det stigende behov for oplysning omkring et broget og meget hurtigt ekspanderende marked. En stor del af det undervisningsprogrammel der udvikles i dag, sælges af små forlag uden de store reklamemuligheder. Det betyder, at oplysning om deres produkt ofte ikke når ud til en større offentlighed. Samtidig står mange lærere i en situation, hvor de har købt datamaskiner, men mangler programmel, som de kan anvende i deres undervisning.

Det er derfor vigtigt, at der - på tværs af forlagsinteresser - findes et samlet katalog, hvor lærere kan søge oplysning om, hvad der findes indenfor deres bestemte fagområde.

Kataloget vil indeholde oplysninger om titel, fagområde, klassetrin, forfat-

ter, forlag, pris, programmeringsprog (hvis program) samt en beskrivelse af programmet eller bogen.

I øjeblikket har vi taget kontakt med de forskellige forlag, og er i gang med at skrive alle de indgåede oplysninger ind på en standardform. Dette arbejde forventes færdigt kort efter sommerferien, og kataloget er derefter klar til udsendelse.

Selve kataloget vil foreligge på to måder:

1. Som et almindeligt katalog, trykt i A4-format, hvor hver beskrivelse fylder en halv side. Kataloget vil være opdelt efter fagområder. Et sådant eksemplar sendes i forbindelse med et nummer af PICCOLINIEN til alle skoler i landet, og er naturligvis gratis.
2. Som en datafil med et tilhørende søge- og sortereprogram. Dette giver brugeren mulighed for at søge oplysninger efter forskellige nøgler såsom forlag, klassetrin, fagområde, forfatter osv. Samtidig kan kataloget benyttes i datalære-undervisningen som et praktisk eksempel på brug af database-system.

Denne version af kataloget anskaffer man sig ved at sende en formateret diskette, samt en frankeret svarkuvert til PICCOLINIEN's redaktion. Det er dog først muligt at rekvirere denne udgave, der kun vil foreligge til PICCOLINE-disketteformater, når kataloget efter sommerferien er udsendt på normal vis.

Kataloget vil jævnligt blive ajourført, og nye udgaver kan rekvireres efter behov.

Så snart kataloget ligger færdigt til distribution, vil dette blive omtalt i PICCOLINIEN.



Fortsat fra forsiden.

## Seymour Papert i Danmark

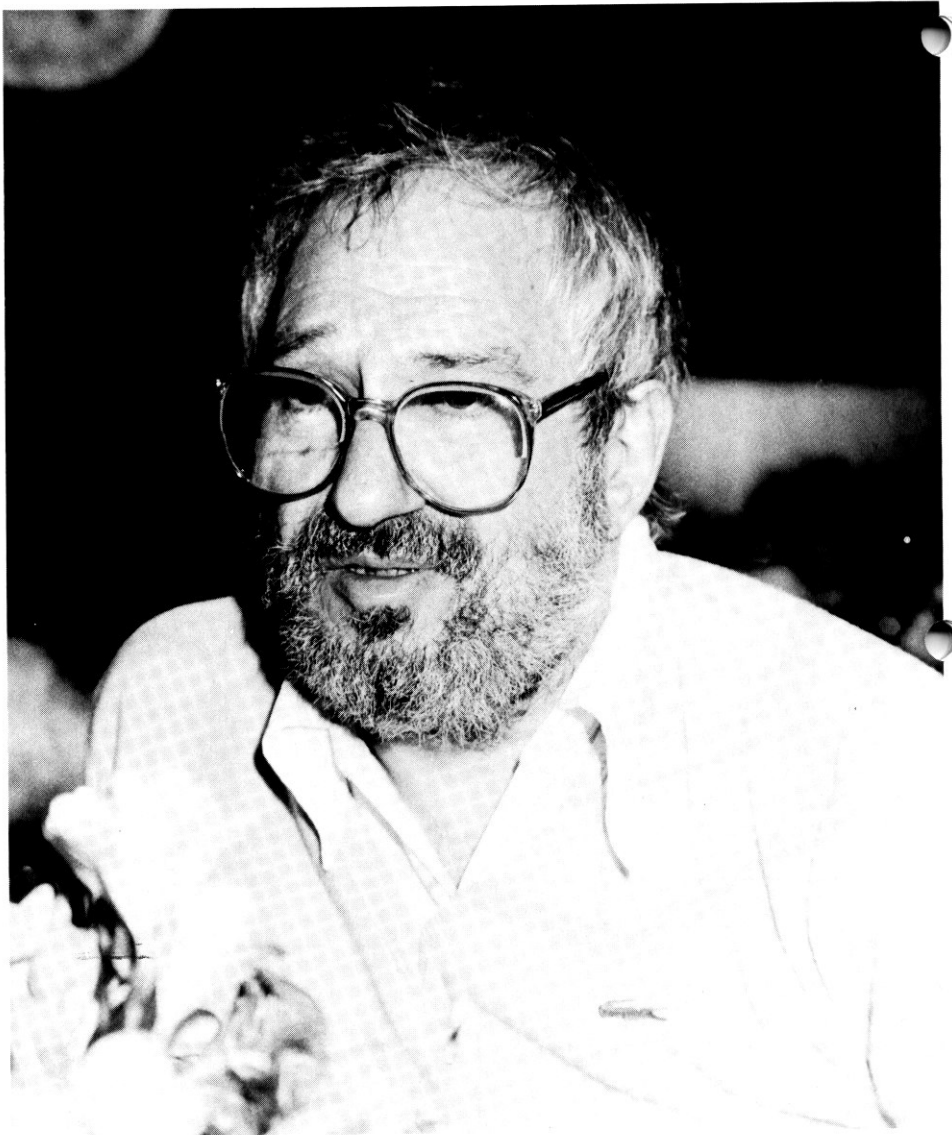
»Allerførst vil jeg gerne slå fast«, siger Seymour Papert, »at vi under ingen omstændigheder kan snakke isoleret om brugen af datamater i skolesystemet. Vi er nødt til at se hvilke muligheder, vi har for at lave nogle grundlæggende ændringer i vort skolesystem, så den almindelige undervisningsform med læreren som den 'alvidende' nedbrydes«.

Han mener ikke, at det er muligt at undervise på traditionel facon i brug af computere, og vender sig meget kraftigt mod den model, man er begyndt på at anvende i folkeskolen i Danmark, hvor datalære bliver et selvstændigt fag.

»Det har vi også prøvet i USA, og med et meget dårligt resultat, idet kun få af eleverne faktisk opnåede nogen færdigheder. Det er vigtigt, at computeren benyttes som et redskab - på linie med andre redskaber som blyant og papir, tavle og kridt, atlas osv. - der kan benyttes, når et specifikt behov opstår, som kan løses ved hjælp af netop dette medie. Den anden vej rundt - at man siger, at vi nu skal lave et projekt, hvor vi benytter datamaten - er helt misforstået. På den måde bliver datamaten aldrig en naturlig del af de arbejdsredskaber, man får i skolen«.

»Det er i den forbindelse vigtigt, at man på de enkelte skoler ikke begynder at oprette egentlige EDB-lokaler, hvis dette kan undgås. Det, at man skal reservere tid til et sådant lokale, og skal bevæge sig udenfor sit eget klasseværelse, gør at muligheden for en spontan anvendelse af datamaten forsvinder«.

Seymour Papert mener, at et EDB-lokale således kommer til at virke begrænsende på elevernes udvikling, idet der fokuseres på maskinerne som et mål i sig selv. Dermed umuliggøres open education', hvor elevernes aktive handlen, eksperimenteren og problemløsning i forbindelse med et projekt er nøgleord. Mange programmer er efter Seymour Paperts mening opbygget efter et meget behaviouristisk undervisningsprincip, og det finder han er et stort problem. Der er dog heldigvis også begyndt at komme flere, der bygger på andre pædagogiske principper.



### COMAL - ikke andet end endnu en BASIC-variant.

»Man skal hele tiden huske på, at datamaten kan bruges indenfor alle pædagogiske retninger, men at den frem for alt vil virke meget forstærkende på disse. På denne måde risikerer vi - hvis vi ikke er opmærksomme på problemet - uden at ville det, at se os selv fanget i en endnu højere grad af polarisering af undervisningsprincipperne«.

### Programmering

Seymour Papert mener dog også, at det er bydende nødvendigt, at man lærer eleverne at programmere, men at dette godt kan foregå som en interaktiv proces, hvor man langsomt igennem skoleårene tilegner sig flere og flere færdigheder indenfor dette område.

»Der er en tendens til, at programmeringskurser er noget, der skal laves i en forfærdelig fart, og så kommer det til at stå som et selvstændigt emne uden relation til alt det andet, der foregår i skolen. Dette vil ikke have nogen mening overhovedet«.

Stillet overfor oplysningen om, at den almindelige holdning i Danmark er, at man ikke skal lære eleverne at programmere, men derimod præsentere dem for en række færdige programmer og anvendelser, for på den måde at forberede dem på den verden og det arbejde de senere kommer ud til, siger Seymour Papert utvetydigt:

»Det er noget vrøvl! ...og det ved I også godt, at det er. Når først eleverne forlader skolesystemet og skal benytte datamater, kommer de til at arbejde med langt mere professionelle og specialiserede systemer, end vi nogensinde vil være i stand til at præsentere dem for i skolen«.

Han mener, at det er langt vigtigere at eleverne f.eks. i forbindelse med simulationer, bliver i stand til selv at lave programmet - eller i det mindste forstå, hvordan det fungerer - og ikke bare tage resultaterne for gode varer. Efter Seymour Paperts mening behøver den slags programmer heller ikke at være indviklede at konstruere, når bare man benytter et fornuftigt programmeringssprog, hvad han dog ikke mener, at man gør i det danske skolesystem.

»Hvad jeg mener er«, fortsætter Seymour Papert, »at et land som Danmark bør - med alle sine ressourcer - være i stand til at udvikle et ordentligt dansk programmeringssprog til undervisningsbrug. Det er helt urimeligt, at springe over, hvor gærdet er lavest, og lade en eller anden påtvinge den samlede danske skoleverden sådan noget som COMAL, der trods alt ikke er andet end endnu en (omend væsentligt forbedret) udgave af BASIC. Det er jo fyldt med ting, der tydeligvis kun er der, fordi man ikke har ofret ressourcer på fra starten at få rensset sproget ud og få lavet et ordentligt arbejde. Lad os se det i øjnene: Hvis man vil have en billig løsning, får man også et resultat derefter.«

Han tilføjer dog, at det naturligvis ikke er noget, der er specielt for Danmark, men at f.eks. de store amerikanske firmaer som IBM, APPLE osv. begår nøjagtigt de samme fejltagelser.

## Egget og hønen

På spørgsmålet om det ikke vil kræve meget store ressourcer at lære alle eleverne at programmere, svarer Seymour Papert:

»Jo, men det gør egentlig heller ikke noget. Det man må se på er, at det ikke er noget, der kan klares isoleret. Hvis man skal undervise i programmering - eller blot anvende en datamat i undervisningen - mens alt andet på skolen er uforandret, så er det faktisk en umulighed. Det kræver nogle ændrede undervisningsforhold, hvor datamaten får mulighed for at indgå som en naturlig del. Men desværre er det historien med ægget og hønen om igen: Ændrede undervisningsforhold er det, der muliggør, at man kan lære at udnytte en datamat, og det at man kan benytte en datamat, er det, der muliggør ændrede undervis-

ningsforhold. Det svære er at finde ud af, hvor man skal starte«.

Seymour Papert mener i den forbindelse, at man skal passe meget på med ikke at sprede datamaterne for meget ud på forskellige skoler. Man bør forsøge at lave meget koncentrede forsøg, hvor man lader en skole få lov til virkelig at benytte mange maskiner, for på den måde at give datamaten en chance for at vise sit værd - eller evt. mangel på samme. Han mener selv, at spredningen af datamaterne ud på mange skoler, hvor hver enkelt skole måske kun får een, er den værste fejl, man har begået i USA i de forgangne år.

## LOGO

Til sidst spurgte vi Seymour Papert om, hvad det efter hans egen mening er, der gør LOGO til et bedre sprog end alle andre.

»LOGO er«, svarer han, »egentlig udsprunget af et andet programmeringssprog kaldet LISP, og har den samme struktur som dette, men en anden og simplere syntaks. Det er denne meget strukturelle opbygning, der gør LOGO og LISP til, hvad de er. I BASIC har forskellige ordretyper helt forskellige udseender: IF-sætninger har een, LET-sætninger en anden osv.«

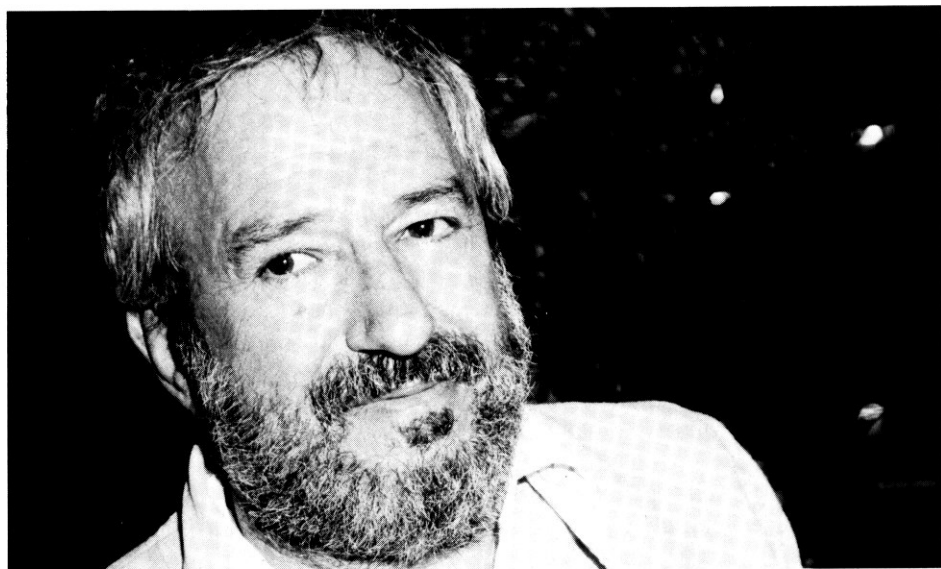
**Datamaten vil virke forstærkende på alle pædagogiske retninger.**

En anden meget markant egenskab er, at LOGO er fuldstændigt procedure-opbygget. COMAL har nærmet sig dette princip ved at tillade procedurer inde i programmet, og er på denne måde en klar forbedring i forhold til BASIC. Men det er stadig en fejl, at man skal lære eleverne at arbejde med programmer, der måske er flere hundrede linier lange, også selvom de er procedure-opdelte.

»LOGO«, fortsætter Seymour Papert, »består af mange små programmer (eller procedurer), der koordineres, men hvor hvert enkelt lille program er en selvstændig enhed, og sjældent fylder mere end nogle ganske få linier«.

LOGO-programmer benyttes i dag på linie med alle andre programmeringssprog. Der findes tekstbehandlingssystemer, budget-programmer, databaseprogrammer osv. skrevet i LOGO, og alle sammen har de den fordel, at de er meget lette at rette i, da strukturen er så enkel.

»Indtil for få år siden«, fortæller Seymour Papert, »var det sådan i USA, at LISP-programmer kun blev benyttet til meget specielle teoretiske forskningsopgaver som f.eks. kunstig intelligens og eksperter-systemer. Men nu har man også fået øjnene op for fordelene ved denne type programmer rundt omkring i erhvervslivet og det offentlige område, således at der nu skrives LISP og LOGO-programmer til næsten alle formål. Dette er en udvikling, som jeg forventer vil fortsætte i de næste år, og som meget hurtigt vil sprede sig til det europæiske marked«.



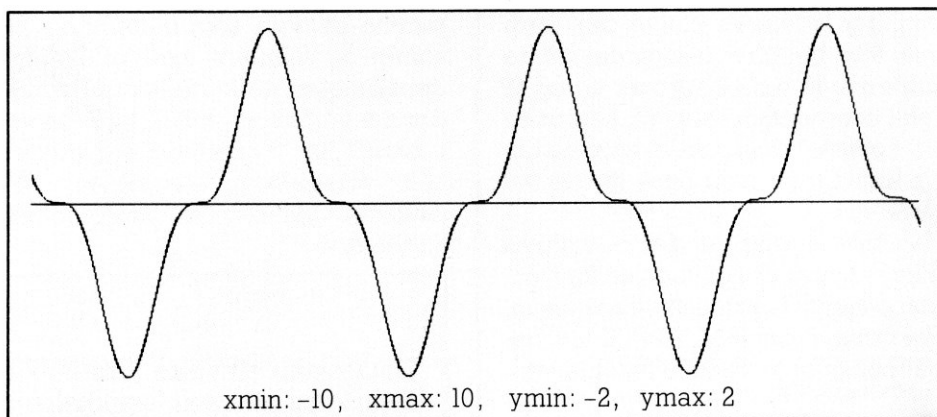
# 603-printeren brugt som plotter

Mange brugere af PICCOLINIEN har anskaffet sig en 603-printer bl.a. for at kunne udskrive grafik-billeder. Man kan benytte printeren til almindelig tekst og grafik umiddelbart, men denne printer har også en lang række special-operationer, der gør den til et avanceret stykke værktøj.

Vi skal i denne artikel se på et eksempel, der illustrerer nogle af de muligheder, man har med printeren. Det afbildede program bruges til at tegne grafen for en vilkårlig funktion - defineret i linie 20. Desuden tegnes også en x-akse.

Dette er i sig selv ikke særligt opsigtsvækkende. Vi har bl.a. her i bladet tidligere vist et langt mere avanceret program, til løsning af en sådan opgave (Feb. 85). Det interessante ved dette program er derimod måden, kurven tegnes på.

I et almindeligt grafisk program bliver hele billedet gemt i en midlertidig fil på disketten, og til sidst skrevet ud linie for linie ved brug af grafiksystemet. Det første man kan bemærke i dette program er, at grafiken *ikke* er indlagt, og at vi derfor selv styrer tegningen fuldt ud. Desuden tegnes punkterne, efterhånden som de udregnes, hvilket betyder, at printeren skal styres, så den kan køre både forlæns og baglæns. Kurven bliver altså tegnet på samme måde, som en plotter ville tegne den - ved at



køre papiret frem og tilbage - eller som man selv ville tegne den på et stykke papir.

For at kunne tegne en graf på denne måde skal vi kunne løse følgende problemer:

1. Vi skal kunne køre både forlæns og baglæns med papiret.
2. Vi skal kunne ændre liniebredden, så lineskift kun bevirker, at vi flytter os, hvad der svarer til een prik.
3. Vi skal kunne nøjes med at skrive en prik ad gangen. Dvs. at antallet af udskrevne tegn bliver væsentlig større end de sædvanlige 80.
4. Vi skal sørge for, at alle tegn - også styretegn som f.eks. lineskift - udskrives umiddelbart, og ikke gemmes i buffere.

Alle disse problemer kan løses ved at sende forskellige styretegn i form af ESC-sekvenser til printeren.

Allerede i sidste nummer var vi inde på at arbejde med disse ESC-sekvenser i forbindelse med opbygningen af specialtegn på 603-printeren. Dette princip løser vores problem nr. 3, idet vi i stedet for som sidst at benytte en tegnbredde på 18 søjler prikker, denne gang kun benytter 1. Herved kan vi skrive een prik, eller en søjle med prikker ad gangen. På 603-printeren vil der så være plads til  $639 = 8 * 80 - 1$  prikker i bredden. (se linie 560-570)

Problem nr. 1 løses ved at sende en af følgende to specialkoder:

CHR\$(27) + "r" - får printeren til at køre baglæns

CHR\$(27) + "f" - får printeren til at køre fremad.

Printeren bliver ved med at køre den vej, den er blevet bedt om, indtil den møder den modsatte besked.



# PICCOLINIEN

(se linie 190 samt 370-450)

Problem nr. 2 løses ved at sende en specialkode, der ændrer linieafstanden. Generelt ser den således ud:

```
CHR$(27)+"Txx"
```

hvor der ved xx skrives et tal på to cifre, der angiver liniebredden målt i 1/144 tomme. Vi sætter liniebredden til det mindst mulige, nemlig 1/144 tomme, og skriver altså:

```
CHR$(27)+"T01"
```

(se linie 120)

Problem nr. 4 løses ved at sende en specialkode, der har udseendet:

```
CHR$(27)+"Æ"
```

(se linie 110)

Programmet indeholder desuden en række begrænsninger:

Man angiver både en maximum- og minimum-værdi for x- og y-akse. Disse værdier angiver grænser svarende til et kvadratisk billede på papiret. Er funktionsværdierne uden for disse rammer, tegnes ingen prikker (se linie 230-360).

Hvis springene lodret mellem de enkelte prikker i grafen er meget store (grafens bliver meget stejl), vil man få brug for at tegne flere prikker ovenover hinanden, så man får en rimelig sammenhængende kurve. Hvor mange prikker, der skal tegnes, styres i linie 460-570. Tegnes kun en enkelt prik, sker det med nål nr. 4 i printerhovedet ud af ialt 8. Herved kan vi tegne søjler, der »peger« såvel fremad som bagud - afhængig af hvilken vej vi kører - ved at benytte henholdsvis nålene 1-3 eller 5-8.

Programmet starter med at tegne x-aksen på det sted, hvor papiret befinder sig. Man skal derfor huske eventuelt at køre papiret fremad inden start, således at der er plads til at køre både forlæns og baglæns.

Programmet er primært ment som en illustration af nogle af de muligheder, man har med 603-printeren, og som en opfordring til selv at forsøge sig med nogle af de andre specialkoder, der kan sendes til printeren.

```
0010 FUNC f(x)
0020   RETURN (SIN(x)'3)
0030 ENDFUNC f
0040 INPUT "indlæs xmin :": xmin
0050 INPUT "indlæs xmax :": xmax
0060 INPUT "indlæs ymin :": ymin
0070 INPUT "indlæs ymax :": ymax
0080
0090 DIM prik$ OF 7
0100 SELECT OUTPUT "printer"
0105 //LINIESKIFT SKER MED DET SAMME
0110 PRINT CHR$(27)+"Æ"
0115 //LINIEAFSTAND SÆTTES TIL 1/144 TOMME
0120 PRINT CHR$(27)+"T01"
0130 lintyk:= (ymax-ymin)/900
0140 bredde:= (xmax-xmin)/639
0150 i:= 1
0160 v:= f(xmin)
0170 lin:= INT(v/lintyk)
0180 FOR j:= 1 TO 639 DO PRINT CHR$(27)+"S0001"+CHR$(16);
0185 //RETNING AF PAPIRFREMFØRING BESTEMMES
0190 IF lin>0 THEN PRINT CHR$(27)+"r"
0200 FOR j:= 1 TO ABS(lin) DO PRINT CHR$(10);
0210 PRINT
0220 FOR j:= xmin+bredde TO xmax STEP bredde DO
0230   v:= f(j)
0240   //BESTEMMELSE AF LINIENUMRENE INDENFOR RAMMERNE
0250   CASE -(v<ymin)+(v>ymax) OF
0260     WHEN -1
0270       tryk:= 0
0280       nælin:= INT(ymin/lintyk)
0290     WHEN 0
0300       tryk:= 1
0310       nælin:= INT(v/lintyk)
0320     WHEN 1
0330       tryk:= 0
0340       nælin:= INT(ymax/lintyk)
0350   OTHERWISE
0360   ENDCASE
0370   //BESTEMMELSE AF RETNINGEN FOR PRINTEREN
      (FREMAD/TILBAGE)
0380   IF nælin>lin THEN
0390     PRINT CHR$(27)+"r";
0400     retn:= 1
0410   ENDIF
0420   IF nælin<lin THEN
0430     PRINT CHR$(27)+"f";
0440     retn:= -1
0450   ENDIF
0460   // BESTEMMELSE AF PRIKHØJDE VED STORE LINIESPRING
      p:= 2^4
0480   IF ABS(lin-nælin)>2 THEN
0490     afst:= ABS(lin-nælin)-2
0500     IF afst>4 THEN afst:= 4
0510     FOR k:= retn TO afst*retn STEP retn DO
0520       p:= p+2^(4-k)
0530     NEXT k
0540   ENDIF
0550   prik$:= CHR$(27)+"S0001"+CHR$(p)
0560   PRINT prik$;
0580   //FREMFLYTNING TIL NÆSTE LINIE
0590   FOR k:= 1 TO ABS(nælin-lin) DO
0600     PRINT CHR$(10);
0610   NEXT k
0620   lin:= nælin
0630 NEXT j
```

# Familie- økonomi- spillet

ved Jan Moth Iversen

Af Poul Andreasen og Sven Thorø.  
Forlaget SIMULTAN 5. udg. 1985.

Materialesæt: 1 grundbog 50 s. + 20 bilag. Pris kr. 50,- pr. sæt. Hertil gratis lærervejledning.

EDB-Program-pakke bestående af en mappe med 1 diskette m. brugervejledning samt 1 materialesæt. Pris ca. kr. 850,-.

Forlaget tilbyder hvert år at revidere programmerne. Pris: 40-50% af nypris.

Spillet er beregnet til anvendelse i 8-10. klasse. Jeg har her brugt det i slutningen af 9. klasse i fagene samtidsorientering/matematik (samlæst). Varighed ca. 20 timer.

Selve spillet går ud på, at eleverne danner familier, som på godt og ondt skal gennemleve et regnskabsår. Inden året kan begynde, må familierne foretage en del valg. Først afgør eleverne, om de ønsker at gifte sig, og om de vil have et barn i familien. Desuden vælger de ud fra de i grundbogen beskrevne muligheder deres arbejds- og boligsituation, evt. anskaffelse af bil, forsikringer m.m. Ud fra disse valg laver familierne forskudsskemaer, lægger et budget, og det egentlige spil kan starte.





Her oplever eleverne via et detaljeret check-regnskab måned for måned familiens økonomiske situation m.h.t. faste udgifter, eventuelle nyanskaffelser samt uforudsigelige hændelser med indvirkning på økonomien. (De sidste snedigt frembragt ved terningkast).

Når regnskabsåret slutter, laver familierne deres selvangivelser med evt. tilhørende ejendomsskema, slutskatten beregnes, og spillet er færdigt. Under hele forløbet har læreren sammen med en eller to elever fungeret som kombineret rådgivning/bank/skattekontor.

Spillet, som man kan køre helt uden brug af de tilhørende EDB-programmer, er efter min mening et meget fint og velgennemtænkt undervisningsforløb. Tonen og hele udformningen virker meget indbydende for eleverne, men det er regnemæssigt drøjt i begyndelses- og slutfasen.

Undervejs i spillet kan man supplere med anvendelsen af EDB-programmerne. Disse er meget let tilgængelige – således tog det kun 10 minutter at uddanne den elev, der var spillets kontormedarbejder, til »EDB-ekspert«.

Pakken indeholder 4 programmer, der kan kaldes fra en hovedmenu:

1. Optagelse af banklån
2. Budgetsimulering
3. Trækprocent – skattekort
4. Slutskatteberegning

ad 1. På et hvilket som helst tidspunkt i forløbet kan en familie optage lån i banken. Programmet erstatter kontorets manuelle udregninger og udfyldelse af lånedokument (som udskrives på printer) – en lettelse af kontorets rutinearbejde. Programmet indeholder 3 niveauer, hvor det sværeste nok er bedre anvendt uden for dette spil.

ad 2. I forbindelse med budgetlægningen kan de enkelte familier arbejde ved maskinerne. Her er der 2 niveauer: et simpelt og et udvidet. Vi arbejdede med det sidste, hvor familierne kan prøve at ændre nogle forudsætninger og få maskinen til at

beregne konsekvenserne af familiens valg. Det er efter min mening det bedste program i hele pakken.

ad 3. Dette program erstatter igen kontormedarbejderens manuelle rutinearbejde m.h.t. udregning af trækprocent og udskrivning af skattekort.

ad 4. Det sidste program annonceres som »et program, der gør den frygtede slutskatteberegning til en leg« – og det virker da også besnærende, når familierne, der har siddet halve timer med den manuelle slutopgørelse, i løbet af ganske kort tid kan få udskrevet en årsopgørelse, selv om den ikke passer med det, de selv regnede ud.

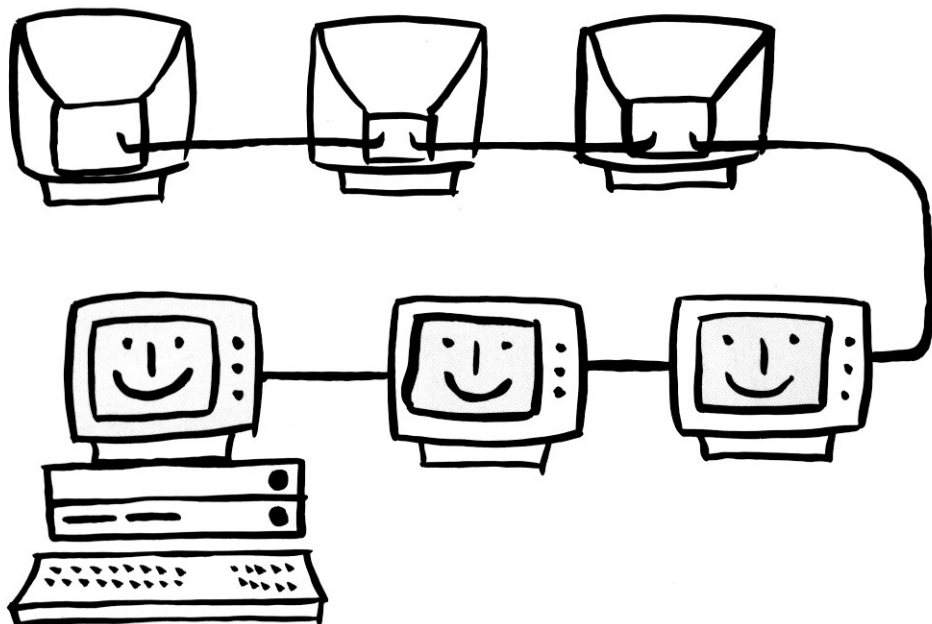
Alt i alt synes jeg, at EDB-programmerne er særdeles anvendelige til spillet – og maskinerne er da også (endnu?) i sig selv en motiverende faktor for mange elever. Jeg går ud fra, at de små kosmetiske fejl på udskrifterne allerede er rettet. De ødelægger helhedsindtrykket.

Imidlertid synes jeg nok, at programmerne kunne forbedres lidt. For ofte virker de blot som super-lommeregner, der spytter et resultat ud, uden at eleverne kan se sammenhængen til de indtastede data. Jeg så gerne flere programmer i stil med budgetprogrammet, hvor eleverne kan hente hjælp – f.eks. til udfyldning af selvangivelsen.

Jan Moth Iversen



## Flere skærme koblet til samme datamat



Mange lærere har været i den situation, at de gerne vil vise et program frem for eleverne, men er stødt ind i den vanskelighed, der ligger i at få ca. 25 elever til at kigge på en 12" dataskærm samtidig. En del har løst problemet ved at anskaffe sig en stor dataskærm, andre har ombygget et fjernsyn til dette formål. Men dels er dette ofte en dyr løsning, og dels kan det - selv på en 26" skærm - stadigvæk være svært at se. Har man mere end en skærm, kan problemet dog ofte løses på en langt simplere og billigere måde, der samtidig er mere effektiv. Man kan nemlig serieforbinde skærmene. Det forudsætter dog, at man har en datamat med en dataskærm, der både har en videoindgang og udgang. Er der en videoudgang, vil der normalt også være en omskifterknap, der kan stå på 75  $\Omega$  eller HIGH.

Man starter med at fjerne videokablerne fra alle skærme, bortset fra den hvorpå man vil køre programmet. Så forbindes videoudgangen på den første skærm med indgangen på den anden, udgangen på den anden med

indgangen på den tredje osv. Til sidst sættes alle omskiftere på HIGH, bortset fra den der sidder på skærmen yderst i kæden.

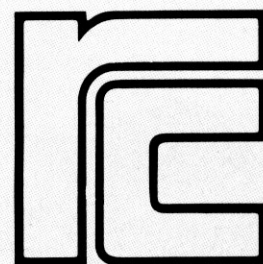
Når maskinen nu startes, vil det være det samme billede, der kommer frem på alle skærmene på en gang, således at ingen får problemer med at se fremvisningen.

Vi har prøvet at forbinde op til 9 skærme med hinanden uden at billedet blev sløret, og flere endnu kan sikkert hægtes på. Der kan dog kræves en lille justering af lysstyrken i billedet.

Hvilke ledninger skal man så benytte?

Det er ledninger, der kan købes i enhver radioforretning, nemlig de såkaldte »japaner-stik«, dvs. små jack-stik. Typisk foreligger de som dobbeltledninger, der dog let kan adskilles.

Prisen er afhængig af længden, men en dobbeltledning på ca. 1,5 meter koster omkring kr. 50,-, og for den pris kan man altså forbinde 3 skærme, mens det vil koste ca. kr. 150,- at forbinde op til 7 skærme.



### Udgiver:

Regnecentralen  
Lautrupbjerg 1  
2750 Ballerup  
Tel.: (02) 65 80 00

### Indlæg fra læserne:

Skal sendes til ovenstående adresse.

### Redaktion:

Mogens Guildal (ansv.)  
Ole Schwander Olsen

### Salgsafdeling:

Lautrupbjerg 1  
2750 Ballerup  
Tel.: (02) 65 80 00  
henviser til nærmeste  
forhandler

### Supportcenter:

Lautrupbjerg 1  
2750 Ballerup  
Tel.: (02) 65 80 00

### Teknisk service:

Glostrup  
Hovedvejen 9  
2600 Glostrup  
Tel.: (02) 96 07 00

Århus  
Klamsagervej 19  
8230 Åbyhøj  
Tel.: (06) 25 04 11

Aalborg  
Limfjordsvej 14  
9400 Nørresundby  
Tel.: (08) 17 80 44

Odense  
Henovej 10  
5270 Odense N  
Tel.: (09) 18 78 15

### Grafisk tilrettelægning og tryk:

Johnsen + Johnsen a/s,  
København