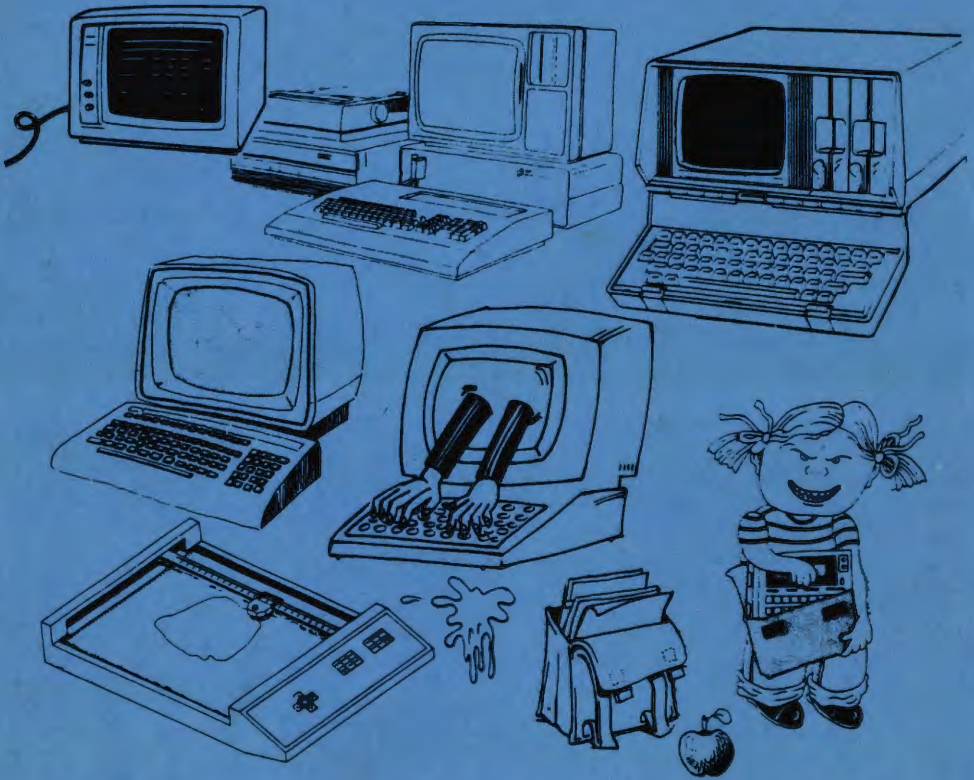


# Dats

---



---

AUGUST 1985 NR. 6

# INDHOLD

<b>Forord</b> .....	3
<b>Datamatassisteret tegnings- fremstilling</b> Ulrich Lysdahl Jensen.....	5
<b>Programmer til plotteren</b> Ole Stausholm.....	25
<b>Edb i undervisningen</b> Karsten Juul-Olsen.....	32
<b>Krav til programmel</b> Karsten Juul-Olsen.....	42
<b>CP/M-spalten:</b> Om submitfiler Karsten Juul-Olsen.....	47
 <b><u>Anmeldelser</u></b>	
Bent Madsen og Arne Mortensen: Computermagt og menneskeværd. ved Niels Brøndum Josephsen.....	49
Kaj Olesen, Nils Hesselberg Møller m.fl.: Datamater i den erhvervsrettede voksenundervisning. ved Bent Rasmussen.....	61
 <b><u>Kurser m.v.</u></b>	
CP/M-kursus.....	63

---

**Ekspedition, abonnements-tegning,  
annoncer:**

TICA, Teknisk Informatik Center Aarhus,  
Aarhus tekniske Skole,  
Halmstadgade 6, 8200 Århus N.  
tlf.: 06 - 16 61 00  
postgiro 7005008

---

**Redaktion:**

Ole Karmark  
Karsten Juul-Olsen  
Niels Brøndum Josephsen  
Kaj Thrysoe

Redaktionen modtager maskinskrevne manuskripter.  
Offentliggjorte artikler honoreres ikke.

---

**Abonnement:**

Dats udkommer med ca. 4 numre årligt.  
Samlet pris for nr. 7, 8, 9 og 10  
inkl. moms og forsendelse: 100 kr.  
Pris i løssalg: 30 kr. pr. stk.

---

## FORORD

Det foreliggende nummer af Dats er det sidste i årgangen 1984 /85.

Nummeret indeholder denne gang en oversigt over de muligheder, der i dag findes for at sammensætte maskinel til et datamatbaseret tegnesystem. Dernæst følger en instruktiv gennemgang af en række procedurer, man kan få brug for under arbejdet med at udføre tegninger på plotteren. I andre artikler opsummeres nogle generelle erfaringer med datalære og datamatformidlet undervisning på de tekniske skoler, og der formuleres nogle brugerkrav til edb-programmel i almindelighed. Den faste CP/M-spalte behandler brugen af submit-filer. Endelig afsluttes nummeret med et par anmeldelser af aktuelle bøger.

Dette nummer markerer samtidig afskeden med Dats i den hidtidige form og opsætning. Den kommende årgang 1985/86 vil stadig bestå af 4 numre, men vi vil bestræbe os på at forbedre såvel form som indhold. Indholdsmæssigt vil vore planer betyde, at numrene fremover gerne skulle blive mere temaorienterede. Således planlægger vi i øjeblikket temaer om bl.a. datamatstøttet undervisning, om datakommunikation og om informationssamfundet. Vi vil belyse disse og andre emner gennem artikler, anmeldelser og som noget nyt: interviews. Vi modtager fremdeles gerne bidrag, læserbreve og anmeldelser fra læserne. Bladet vil fremover også bære præg af, at vi vil sigte på at få en bredere målgruppe i tale. Debatten om informationsteknologi til undervisningsbrug bør ikke indsnævres og styres af en for snæver teknisk interesse. Vi har på de tekniske skoler undervist i datalære i en halv snes år efterhånden, og vi vil gerne bidrage til en fordomsfri erfaringsudveksling mellem bredere kredse i undervisningssektoren. Netop i denne tid starter andre undervisningsinstitutioner med dataundervisning.

Nye lærerkræfter med andre indgangsvinkler ser på tingene med friske øjne. Hvordan griber man indførelsen af edb/informatik an i Folkeskolen, i Gymnasiet? Hvilke tanker gør man sig? Dats ser det som sin opgave at formidle erfaringer, nye ideer, kritisk indsigt, også på tværs af forskellige skoleformer og faggrænser.

På formsiden vil vi tilstræbe at give bladet et mere indbydende layout. Det skulle bl.a. betyde en bedre trykkvalitet, flere illustrationer, og et mere brugervenligt format. Alt dette kan selvfølgelig ikke undgå at berøre abonnementsprisen. Den hidtidige uhørt lave introduktionspris for sidste årgang kan ikke opretholdes. Vi har set os nødsaget til at hæve prisen til 100 kr. for årgangen 1985/86.

## DATAMATASSISTERET TEGNINGSFREMSTILLING

Ulrich Lysdal Jensen, ICL

I fortsættelse af en tidligere artikel her i bladet om emnet CAD/CAM - datamatassisteret konstruktion og produktion - skal vi i denne artikel se nærmere på datamatassisteret tegningsfremstilling, som udgør en del af hele CAD/CAM-komplekset.

For at kunne fremstille konstruktionstegninger med hjælp (ikke ved hjælp) af en datamat, skal man have adgang til to ting, nemlig noget maskinel og noget programmel.

I denne artikel vil vi gennemgå maskinellet og gemme det i og for sig endnu vigtigere programmel til et senere indlæg.

Det har indtil for nylig kun været meget store datamater, der har været på tale i forbindelse med CAD/CAM og tegningssystemer. Indenfor de sidste par år har det imidlertid vist sig, at også mini- og mikrodatamater kan gøre god fyldest som hjælpemidler i forbindelse med datamatassisteret tegning, konstruktion og produktion.

I virkeligheden er et godt tegningssystem ikke nødvendigvis betinget af en stor kraftig datamat. Et godt tegnesystem er karakteriseret ved, at det er let for brugeren at anvende uden noget større kendskab til datamater og deres funktion i øvrigt. Naturligvis kan man udnytte en stor datamats kapacitet til at udvikle en høj grad af brugervenlighed, men ikke desto mindre er der blevet lavet en del tegnesystemer på større datamater, som set fra et brugersynspunkt er klodsede og vanskelige at arbejde med. På den anden side er der på det seneste fremkommet flere særdeles fikse tegneprogrammer til mikrodatamater.



Selv til de mindste mikrodatameter - de såkaldte 8-bit mikroer - findes der tegneprogrammer, der i praksis har vist sig at tilgodese et behov indenfor en absolut rimelig investering.

Men hvorfor skulle ikke også tegnesystemer følge den generelle 'lovmæssighed', at et system udnytter den tid/kapacitet, der er til rådighed. På store datamater fristes man således til at indføre så mange faciliteter og muligheder, at det ender med at blive uoverskueligt for alle andre end netop dem, der selv har fremstillet systemet. Mikrodatamater sætter en vis grænse for, hvor meget der kan proppes ind i et og samme system, og derfor bliver resultatet oftest, at mikrodatamatsystemerne er lettere at lære og håndtere.

Så mange ord for at understrege, at der i det følgende ikke opereres med millioninvesteringer. En virksomhed kan i dag indføre datamatassisteret tegningsfremstilling for en investering på under 100.000 kroner. Med en investering i denne størrelsesorden kan man komme igang og indhøste nogle erfaringer om, hvorledes systemer af denne art fungerer indenfor virksomhedens rammer. Viser det sig, at virksomheden kan have glæde af at satse på CAD/CAM, kan man jo altid sidenhen udvide systemerne.

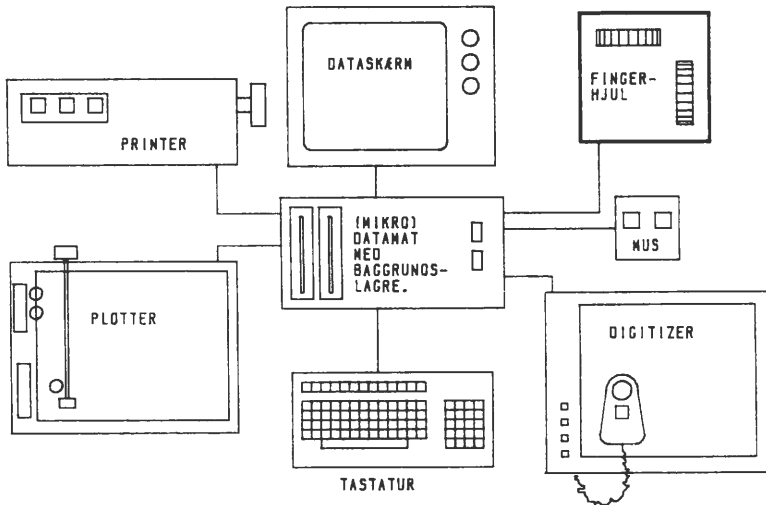
Viser det sig omvendt, at udbyttet er tvivlsomt, er det også til at overskue skaden ved den foretagne investering. Hvem har lyst til at indrømme, at en investering i et system til nogle millioner var en fejlinvestering - og derefter tage konsekvensen heraf?

Nok om det. Vi skal nu beskrive de maskinelle komponenter, der indgår i et datamatbaseret tegnesystem.

Vi skal naturligvis anvende en datamat, som skal være udstyret med et baggrundslager til opbevaring af selve tegnesystemet

samt til arkivering af de tegninger, der fremstilles ved hjælp af dette. Datamaten kan som nævnt udmærket være en mikrodatamat, og baggrundslagrene skal blot være store nok til at dække det aktuelle behov. Dvs., at man kan anvende disketter eller faste disk efter lyst og behov. Mikrodatamatsystemer af denne type er sikkert læserne bekendt, så derfor vil vi ikke her spille plads på at gøre rede for denne side af sagen.

Derimod vil vi se nærmere på nogle af de andre komponenter, der kan/skal indgå i et tegnesystem som skitseret på nedenstående figur.





Vi kan dele komponenterne op i tre kategorier:

- Der skal være mulighed for at instruere datamaten om udseendet af den tegning, der ønskes fremstillet. Dette kan vi kalde at inddatere - eller mere uhøjtideligt at tegne - en tegning. Til dette brug skal vi have adgang til et inddateringsmedium - tegneredskab -, som kan være et tastatur, en digitizer og meget andet som diskuteret nedenfor.
- Man skal gerne kunne se, hvorledes tegningen tager sig ud, når man arbejder med den. Det være sig, medens man er ved at fremstille tegningen, eller hvis man henter den frem fra arkivet for at rette i den. Hertil er en dataskærm med gode grafiske egenskaber velegnet.
- Uanset den megen edb-teknik er der i langt de fleste tilfælde et behov for at fremstille en papirtegning til brug ude i produktionsleddet. Man skal naturligvis ikke forsværge, at der en dag står dataskærme i skurvognene eller på stilladset. Så længe det imidlertid endnu er almindeligst, at håndværkere, arkitekter, ingeniører osv. arbejder med papirtegninger i marken, er der behov for plottere og printere til fremstilling af papirkopier af de tegninger, der er fremstillet med assistance fra datamaten.

## **DATASKÆRME**

Man kunne spørge: Hvorfor skal en tegning først vises på en dataskærm? Hvorfor kan man ikke straks få tegningen skrevet ud på papir via en printer eller plotter? Svaret må være, at naturligvis kunne man det, men ved først at opbygge en tegning på en dataskærm, har man gode muligheder for at slette og rette i tegningen, indtil man er tilfreds med resultatet. En

dataskærm slides ikke på samme måde ved sletninger som et stykke papir, der udsættes for viskelæder eller raderkniv. Altså kan man opnå en langt bedre kvalitet af den endelige papirtegnning, når tegningen først redigeres på en dataskærm.

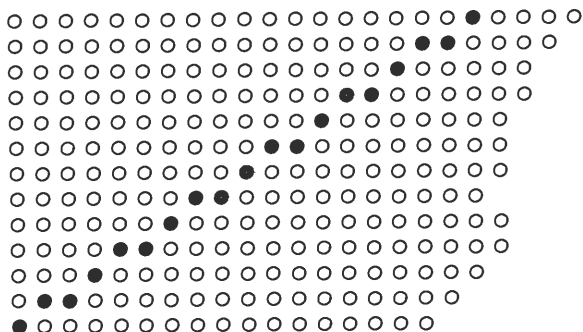
Der findes to typer dataskærme, der er velegnede til at vise tegninger på:

- Vektorskærme
- Rasterskærme

Vektorskærme er den type, der kommer papir og blyant nærmest. En vektorskærm fungerer på den måde, at en elektronstråle 'brænder' et spor i et fosforlag på skærmens (glaspladens) bagside. Elektronstrålen kan bevæge sig i alle mulige retninger og derved tegne rette linjer og kurver. Det spor, elektronstrålen afsætter i fosforlaget bliver stående i nogen tid (mange minutter) og kan kun fjernes helt ved at hele skærmen renses. Fra tid til anden må hele billedet på vektorskærmen friskes op - gentegnes. Hvis man skal slette en del af en tegning, må man slette hele tegningen og derefter gentegne den ikke slettede del. (Dette udføres naturligvis automatisk af datamaten, som har opbevaret informationer om tegningen i arbejdslageret eller på baggrundslageret).

Vektorskærme tegner meget hurtigt, men på grund af de begrænsede muligheder for at slette og behovet for at friske tegningen op, kan de i praksis være langsomme at arbejde med. Hertil kommer, at vektorskærme er dyre at anskaffe og ved flittig anvendelse skal selve skærmen ofte udskiftes. Alt i alt er resultatet, at vektorskærme kun er lidt udbredt og kun i forbindelse med større systemer.

Rasterskærme er den type skærme, der også anvendes til sædvanlige alfanumeriske informationer. På en rasterskærm består skærmfladen af et stort antal punkter arrangeret i lodrette og vandrette rækker. En afbildning på skærmen dannes ved, at det enkelte billedpunkt defineres som havende en vis farve. Hvert enkelt punkt - hvis fysiske udstrækning er nogle få tiendedele millimeter - kan enten være lysende eller mørkt. Eventuelt kan det enkelte punkt være lysende med flere forskellige intensiteter (gråtoner) og/eller farver. Som man vil kunne se ved at studere f.eks. et avisbillede, kan man opbygge et billede - et rasterbillede - ved hjælp af sådanne punkter. Med rastergrafik må en skrå linje vises tilnærmet, med "trappetrin". Har raster-skærmen imidlertid en rimelig høj opløsningsgrad (målt ved antallet af punkter) kan sådanne tilnærmelser let opfattes af øjet som rette linjer. For fuldstændighedens skyld skal det understreges, at en ret linje, der på en rasterskærm vises ved en tilnærmelse, naturligvis vil blive tegnet på en plotter som en 'rigtig' ret linje.



Skærbilledet på en rasterskærm skabes også ved hjælp af en elektronstråle, der oplyser punkter i et fosforlag, men i modsætning til vektorskærmen, beholder et punkt kun sin oplyst-  
hed i en brøkdæl af et sekund. Derfor genopfrisker elektronstrå-  
len - ligesom på en almindelig TV-skærm - skærbilledet adskil-  
lige gange pr. sekund. Det er derfor også nemt at slette helt  
ned til et enkelt punkt på skærmen med øjeblikkelig virkning.

## **INDDATERINGSSEHEDER**

Den/de metoder, der anvendes til inddatering af en tegning til  
en datamat, har stor betydning for brugervenligheden af et  
tegneseystem.

På den ene side skal inddateringsmetoden helst være let at  
tilegne sig, hvilket den bliver, jo mere den svarer til det at  
benytte en blyant. På den anden side må en bruger af et datamat-  
assisteret tegneseystem ikke holde så stædigt på at ville have  
en pen i hånden, at væsentlige muligheder derved udelades.

Med den teknik, der er tilgængelig i dag, kan man meget vel  
fremstille tegninger på en datamat ved at tegne med en pen på  
et eller andet underlag. Imidlertid kan man ofte i forbindelse  
med konstruktioner - f.eks. en tangent til en cirkel, en kurve  
gennem et antal punkter osv. - få datamaten til at udføre det  
trivielle konstruktionsarbejde og de dermed forbundne beregning-  
er, hvis man giver datamaten de nødvendige instrukser på en  
anden måde - f.eks. via et tastatur.

Mange tegneseystemer åbner mulighed for at kombinere inddatering  
fra to eller flere medier - f.eks. tastatur og digitizer.

Tastaturet er tæt knyttet til en dataskærm og er derfor det første medium, man kan tænke på i forbindelse med inddatering af tegninger.

Hvis man - som i de fleste tegnesystemer - opfatter dataskærmen som et koordinatsystem (hvor f.eks. hvert punkt på en raster-skærm svarer til et koordinatpunkt i et to-dimensionalt koordinatsystem) kan man inddatere de enkelte tegningselementer ved at indtaste koordinatsæt. En ret linje kan inddateres ved at indtaste koordinaterne for linjens to endepunkter el. lign. En cirkel kan beskrives ved at indtaste koordinaterne for centrum samt længden af radius osv. Det kræver ikke den store fantasi at forestille sig det arbejde, der vil være forbundet med at beskrive alle tegninger til et højhus eller for den sags skyld et almindeligt parcelhus på denne måde. Men det er en farbar vej. Metoden er til gengæld meget nøjagtig og foretrækkes af mange i forbindelse med detaljkonstruktioner, der kræver stor præcision.

Noget andet, man gerne skulle kunne udføre ved datamatens hjælp, er at transformere dele af en tegning. Hvis man f.eks. ønsker at rotere en bestemt tegningsdetalje et givet antal grader mod uret, kan det godt være noget af et arbejde via et tastatur at beskrive, hvilken tegningsdel, der er tale om.

En anden og mere relevant anvendelse af tastaturet i forbindelse med datamatbaserede tegnesystemer er at vælge de funktioner, man ønsker udført, ved hjælp af taster på tastaturet. De fleste tegnesystemer er udformet således, at man vælger mellem forskellige funktioner - f.eks. 'Tegn ret linje', 'Roter tegningsdel', 'Flyt tegningsdel', 'Slet', 'Skriver' etc.

Valg af sådanne funktioner kan være knyttet til bestemte taster på tastaturet. F.eks. kan funktionen 'Tegn cirkel' være

knyttet til 'C' eller 'O' (dejlig rundt). Man kunne vove den påstand, at et godt tegnesystem ikke burde omfatte flere funktioner, end det antal, der kan vælges fra tastaturets taster. Hvis der yderligere er gjort noget ud af at knytte de enkelte funktioner og taster sammen på en måde, der er let at huske (f.eks. et O for cirkel), er man godt på vej mod den tidligere omtalte brugervenlighed.

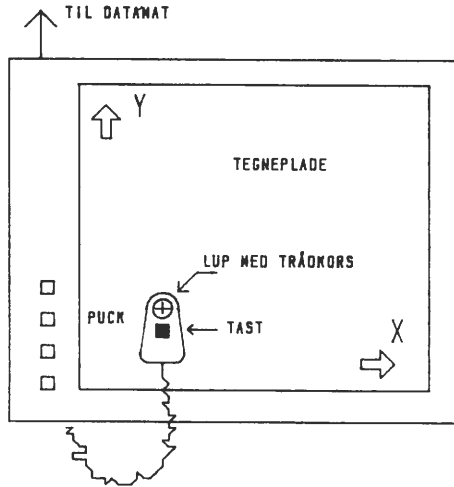
I almindelighed vil et tastatur af sædvanlig udformning, dvs. som et skrivemaskinetastatur, være tilstrækkeligt i forbindelse med tegnesystemer. Enkelte tegnesystemer benytter sig dog af specielle tastaturer. Der kan f.eks. være tale om en helt speciel placering/udformning af tasterne. En anden specialitet kan være, at tasterne er forsynet med lysindikatorer, som hjælper brugeren med at finde frem til de rette taster.

Digitizer En digitizer består af en tegneplade, hvorpå man kan bevæge en pen eller en flad tingest kaldet en puck. Tegnepladen er at opfatte som et todimensionalt koordinatsystem, og fra digitizeren sendes informationer til datamaten i form af koordinater, der fortæller, hvor på tegnepladen pennen/pucken befinder sig. På denne måde kan punkter i et koordinatsystem på dataskærmen udpeges via digitizeren.

Digitizerens pen/puck vil endvidere være forsynet med en kontakt. For pennens vedkommende er kontakten sædvanligvis anbragt i pennens spids, således at man aktiverer kontakten ved at trykke pennen mod tegnepladen, medens pennen bevæges. Puckens kontakt vil være en tast a la en tastaturtast anbragt oven på pucken.

Sammen med hvert sæt koordinater sendes en kode, der fortæller, om kontakten er tændt eller slukket. I datamaten kan denne

information anvendes til at afgøre, om brugeren ønsker at tegne i det punkt, der udpeges. Alternativet kan være, at der blot skal vises en markør i det udpegede punkt.



Der findes også pucker, der er forsynet med 2, 4 eller endnu flere taster, som hver for sig sender en bestemt kode sammen med koordinatsættene. Flere forskellige koder kan i tegnesystemet anvendes til f.eks. at bestemme farve på det, der tegnes på skærmen.

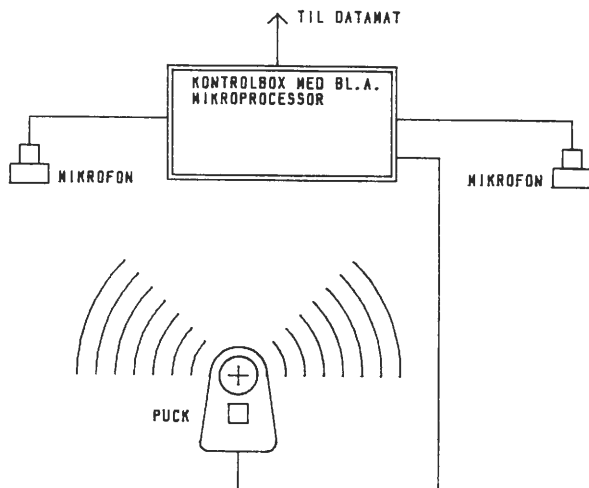
Man arbejder altså med en digitizer på den måde, at man på tegnepladen udpeger de koordinater, der ellers skulle indtastes fra et tastatur. På denne måde er det f.eks. også let at overføre eksisterende tegninger til datamaten, idet man blot placerer den foreliggende tegning på tegnepladen, hvorefter man tegner af. Man siger, at man omsætter tegningen til digital information - digitaliserer tegningen. Deraf navnet digitizer.



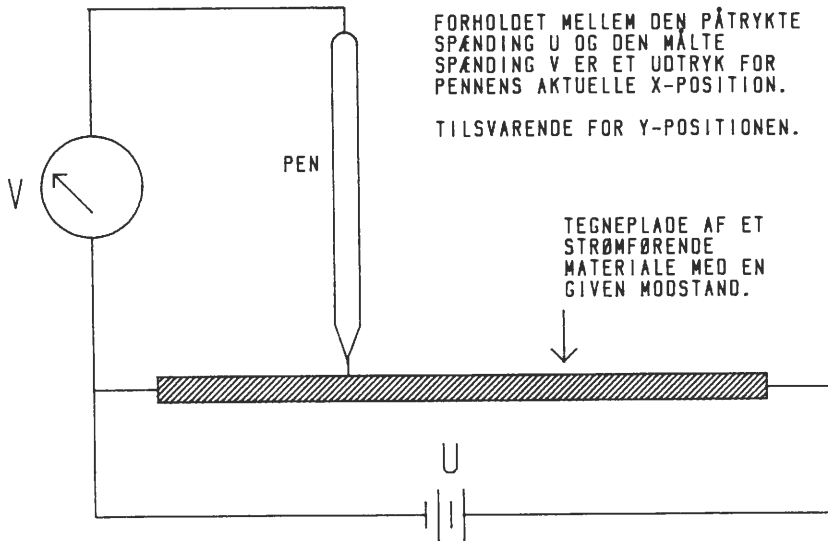
En digitizer kan være opbygget efter mange forskellige principper. I de fleste tilfælde udsendes der et eller andet signal fra pennen/pucken. Dette signal opsamles i tegnepladen, hvorfra det sendes til en mikroprocessor i selve digitizeren. I mikroprocessoren omsættes signalet til et koordinatsæt, som derefter sendes via et kabel til datamaten.

Det signal, der udsendes fra pennen/pucken kan f.eks. være et magnetisk felt, der i tegnepladen opfanges af et net af tråde, som opfatter hhv. X- og Y-retningen. En anden mulighed for at opfange magnetiske signaler i tegnepladen er at udnytte det fænomen, at visse metaller - f.eks. nikkel - deformeres af en magnetisk påvirkning.

En anden form for digitizer er baseret på ultralyd, idet pennen/pucken udsender ultralyd, som opfanges af to mikrofoner. Herved kan man måle tidsforsinkelsen for lyden mellem pen/puck og de to mikrofoner. Denne information kan i en mikroprocessor omregnes til koordinater. Fordelen ved denne type digitizer er, at der ikke kræves en speciel tegneplade. Man kan tegne direkte på en bordplade. Endvidere kan man ved at indføre en tredje mikrofon skabe en 3-dimensional digitizer.



En tredje mulighed er at anvende en tegneplade, der udgør en modstand i et kredsløb, hvor man så med pennen kan måle en spændingsdeling for hhv. X- og Y-retningen. Med dette princip kan man fremstille meget små digitizere.



Uanset hvilket princip der anvendes, er opløsningen for en digitizer, dvs. den mindste afstand mellem to koordinatpunkter, af størrelsesordenen 0,1 mm.

Mus. En såkaldt mus ligner i sin udformning meget pucken fra en digitizer, men musen kræver ikke nogen speciel tegneplade. Musen placeres direkte på en bordplade el.lign. Når musen bevæ-

ges en given afstand (f.eks. et par millimeter), sendes en impuls til datamaten, som beskriver bevægelsens retning. Denne information kan i datamaten anvendes til f.eks. at flytte en skærmmarkør en enhed (et punkt) i den angivne retning. Musen er ligesom pucken forsynet med en eller flere taster, der kan aktiveres, når man ønsker f.eks. at starte tegning, eller at markere et punkt, der skal indgå i en konstruktion.

Musen minder i sin funktionsmåde meget om piltasterne på et tastatur, og mange gange er de signaler, der sendes fra en mus til datamaten de samme, som sendes fra tastaturets piltaster. Ved en enkelt aktivering angiver man en enheds flytning i en af de fire retninger 'op', 'ned', 'højre' eller 'venstre'.

Der findes en række andre inddateringsmedier, der fungerer efter samme princip som musen:

- Joy stick: en gearstangsagtig indretning, som kan bevæges op, ned og til siderne.
- Fingerhjul: to hjul, der kan aktiveres ved hjælp af en finger, og som angiver hhv. bevægelsen op-ned og højre-venstre.
- Kugle: en metalkugle, der kan bevæges ved hjælp af håndfladen.

Som nævnt kan en digitizer anvendes til inddatering af eksisterende tegninger. En anden metode til overførsel af eksisterende tegninger til en datamat er skanning. Her bevæges en lysstråle (f.eks. infrarødt lys) frem og tilbage over tegningen. Udfra det reflekterede lys bestemmes, om et givet punkt er mørkt eller lyst. På denne måde kan man få registreret en tegning som lyse og mørke punkter (ligesom på rasterskærmen).

Skanning kan ske enten ad mekanisk eller elektronisk (TV-kamera) vej. Sidstnævnte er den hurtigste og dyreste.

Et problem i forbindelse med skanning er, at tegningen kun er registreret som punkter. Man kan altså ikke se, om et givet punkt udgør en del af en ret linje eller en cirkel. Dette betyder begrænsede muligheder for at arbejde videre med tegningen - f.eks. lave forstørrelser, udplotning osv. Derimod kan en skannet tegning udmærket - evt. efter at der er slettet og tilføjet i den - gengives på f.eks. en matrixprinter (se senere). Skanning kan eksempelvis anvendes til at gøre dårlige fotokopier gode.

## UDSKRIFTSSENHEDER

Efter at have fremstillet og redigeret en tegning på en data-skærm, vil man som nævnt oftest ønske denne tegning udskrevet på papir. Dette kan ske på to måder, enten i form af en stregtegning, der fremstilles ved hjælp af en plotter, eller i form af en rastertegning, der kan fremstilles ved hjælp af en matrixprinter.

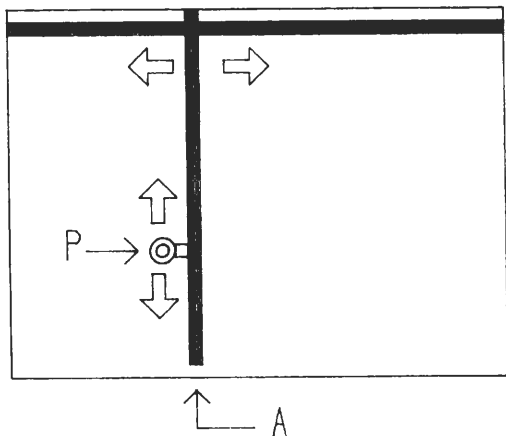
Plotter. En plotter til fremstilling af tekniske tegninger er sædvanligvis af typen kendt som en X-Y-plotter. Navnet hentyder til, at plotteren er i stand til at tegne rette linjer i X-retningen eller vinkelret herpå i Y-retningen. Når der her tales om linjer, menes meget korte linjer. For de fleste plottes vedkommende er den korteste linje, der kan tegnes på 0.1 mm. Ved sammensætning af mange sådanne linjesegmenter kan plotteren bringes til at tegne meget nøjagtige linjer, cirkler, kurver m.v.

Det, der skal til for at sammensætte linjesegmenter til større helheder, vil være indbygget i plotteren, således at denne kun skal have instruks (fra datamaten) om at tegne f.eks. en cirkel. Derefter sørger plotterens automatik selv for at opbygge cirklen af et større antal rette linjer.

Selve tegningen - at en pen tegner en streg på et stykke papir - kan foregå på to måder:

- Pennen kan bevæge sig i forhold til papiret, som holdes fast på et underlag. Pennen vil da typisk være placeret i en slæde, der ved hjælp af stepmotorer kan bevæges i step af f.eks. 0,1 mm længde i hhv X- og Y-retningen. Denne form for plottere betegnes også 'flat bed' plottere, fordi papiret er anbragt fladt på et underlag.

#### FLAT-BED PLOTTER.

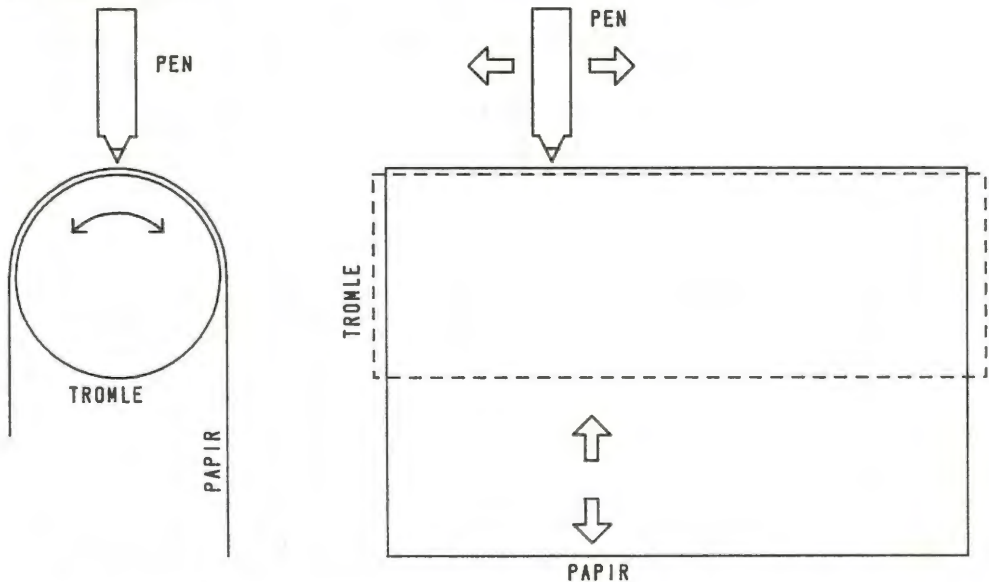


1. BEVÆGELSE AF ARMEN A GIVER BEVÆGELSEN I X-RETNINGEN.
2. BEVÆGELSE AF PENNEN P LANGS ARMEN A GIVER BEVÆGELSEN I Y-RETNINGEN.

PENNEN P KAN SÆNKES ELLER LØFTES ALT EFTER OM MAN ØNSKER AT TEGNE ELLER BLOT FLYTTE TIL EN NY POSITION.

- Pennen kan sidde fast, medens papiret bevæges under pennen. Nu findes der ikke mange plottere, der udnytter denne metode rendyrket. Derimod findes der de såkaldte tromleplottere, hvor papiret er fastgjort til en tromle, der kan dreje rundt og dermed give bevægelsen i den ene retning - f.eks. X-retningen. Bevægelsen i den anden retning opnås ved at lade pennen bevæge sig langs en akse, der er parallel med tromlens.

### TROMLE PLOTTER.



En egenskab, der karakteriserer en plotter, er størrelsen af den tegning, der kan fremstilles ved hjælp af plotteren. Plottere, der kan klare tegninger på op til A3-format, vil oftest være af typen 'flat bed'. For tegningsformater større end A3 vil man finde, at dyrere plottere er af typen 'flat bed', medens plottere i den billigere ende af prisskalaen ofte er tromleplottere.

En væsentlig fordel ved en tromleplotter er, at den i forbindelse med større tegningsformater fylder mindre end 'flat bed' plotteren, som jo nødvendigvis må have et fladeareal mindst svarende til tegningsformatet.

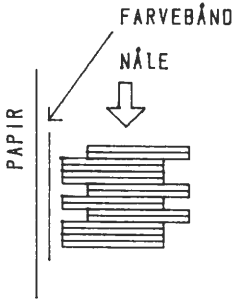
En anden vigtig faktor ved fremstilling af stregtegninger er pen og papir. Kvaliteten af en i øvrigt omhyggeligt fremstillet tegning kan blive elendig, hvis man anvender den forkerte kombination af pen og papir. På den anden side kan lidt omhu på dette område føre til meget professionelle resultater.

Mange plottere leveres i dag med et udvalg af filtpenne i flere farver samt med den mulighed, at man kan instruere plotteren om at skifte pen under tegnearbejdet. Et antal forskellige penne kan anbringes i et magasin, hvori plotterens penholder kan parkere og hente penne. På denne måde kan man altså indlede sit bekendtskab med en ny plotter med at tegne farvestrålende tegninger med brug af det medfølgende udvalg af filtpenne. Når man er blevet træt af denne sport, kan man begynde at overveje, om der er andre former for tegnepenne, der kan komme på tale. Det mest professionelle resultat opnås utvivlsomt ved brug af rørpenne, der tegner med tusch. Med den rette kombination af tusch, penne og tegnepapir af rette kvalitet kan man opnå forbløffende resultater med selv prisbillige plottere. Muligheden for at skifte pen under tegnearbejdet kan naturligvis i tusch-tilfældet udnyttes til at arbejde med flere forskellige stregtykkelser. Der er imidlertid mange opgaver, hvor en filtpen gør fyldest. Den er jo nemmere og mere renlig at arbejde med end tusch.

En speciel form for plottere er fotoplottere, hvor pennen er en lyspen, der tegner på lysfølsom film. Sådanne plottere anvendes bl.a. til fremstilling af de film, der anvendes ved fabrikation af elektroniske kredsløbskort.



**Matrixprintere.** Mange af de printere, der i dag anvendes i forbindelse med datamater til udskrivning af almindelig tekst er af typen matrixprintere. Sådanne printere skriver ved hjælp af et antal lodrette nåle, der kan forskydes i forhold til hinanden, og som kan presses mod et farvebånd, der afsætter et prik-mønster på et stykke papir. Ved udskrift af almindelig tekst opbygges bogstaver, tal og andre skrifttegn ved hjælp af prik-mønstre (rastere eller matricer) på samme måde, som det sker på en raster dataskærm.



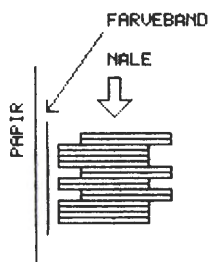
1. DE ENKELTE NÅLE INDSTILLES.
2. ALLE NÅLE SLÅS MOD FARVEBÅND/PAPIR.
3. NÅLENE RYKES EN NÅLEDIAMETER MOD HØJRE OG PROCESSEN GENTAGES.

Med de fleste matrixprintere har man endvidere mulighed for fra datamaten at styre den enkelte nål, og det er da ikke svært at få matrixprinterens til at gengive de lysende punkter på en raster dataskærm som sorte prikker på et stykke papir.

En tegning fremstillet på denne måde kan ikke i kvalitet helt måle sig med en tegning, der er udtegnet på en plotter. Alligevel er kvaliteten af en matrixprinter-tegning tilstrækkelig til mange formål, nemlig både fremstilling af tegninger og udskrivning af tekst. Hertil kommer, at det oftest er hurtigere at udskrive en tegning på en printer end på en plotter.

Nogle matrixprintere kan arbejde med flerfarvede farvebånd, således at man kan fremstille farvelagte tegninger på sin printer.

Alle tegninger i denne artikel har hidtil været udført med en plotter. Her følger til sammenligning en matrixprinter-tegning.



1. DE ENKELTE NALE INDSTILLES.
2. ALLE NALE SLÅS MOD FARVEBAND/PAPIR.
3. NALENE RYKKE EN NALEDIAMETER MOD HØJRE OG PROCESSEN GENTAGES.

Andre typer printere, der kan anvendes til fremstilling af papirkopier af tegninger fremstillet på en dataskærm, er:

- Laserprintere, som brænder spor i papiret ved hjælp af en laserstråle. Med en laserprinter kan man opnå kvalitetskopier, og laserprinterens kan også efterhånden prismæssigt konkurrere med matrixprintere og plottere.
- Ink-jet printere skriver ved hjælp af meget tynde blækstråler. Blæksprøjten på en ink-jet printer kan ofte udskiftes (automatisk eller manuelt), således at man også i dette tilfælde kan fremstille flerfarvede tegninger.

Endelig findes der til visse dataskærme muligheden for at tilkoble en art fotokopimaskine, der kan frembringe en papirkopi af den tegning, der vises på skærmen.

## KONKLUSION

Det foranstående har været en opremsning af nogle af de muligheder, der i dag findes for at sammensætte maskinel til et databaseret tegnesystem. Gennemgangen, som næppe er fuldstændig, skulle gerne vise, at der er mange muligheder indenfor de enkelte delområder. Mulighederne varierer med hensyn til kvalitet, pris og arbejdshastighed. Når man derfor skal vælge tegneudstyr, gør man klogt i nøje at overveje sine behov og vælge udstyr i overensstemmelse hermed.

Ved valg af udstyr, skal man ikke blot tage udgangspunkt i, hvad man gør i dag. Man skal i højere grad se på, hvad der vil ske, når man har anskaffet udstyret.

Som et godt eksempel herpå kan nævnes tegningsformater. En af de erfaringer, man ofte vil gøre efter indførelsen af datamat-assisteret tegningsfremstilling er, at behovet for store tegningsformater reduceres eller helt bortfalder. Dette hænger sammen med, at man med datamatens hjælp hurtigt udfra en oversigtstegning kan fremstille detaljtegninger. Det drejer sig kun om at instruere datamaten om at fremstille en bestemt detalje i et nyt målestoksforhold. Derfor skal man ikke gå ud fra som givet, at ens fremtidige plotter nødvendigvis skal tilgodese behovet for de samme tegningsformater, man arbejder med under den manuelle tegningsfremstilling. Det er da også meget nemmere at stå med en A3 tegning på et stillads eller i en fabrikshal fremfor at skulle kæmpe med A1 eller måske endog A0 formater.

Tager man ovenstående med i betragtning ved anskaffelse af tegneudstyr, vil man sikkert også indse det hensigtsmæssige i at starte med en mindre investering, indtil man har indhøstet nogle erfaringer.

## PROGRAMMER TIL PLOTTEREN

**Ole Stausholm, Aarhus tekniske  
Skole**

---

Det kan ofte i undervisningssituationer være en motiverende oplevelse for eleverne at arbejde med plotterprogrammer. At man kan formulere og beskrive en række geometriske figurer med hjælp af funktioner eller linjer, for derefter at beskrive det som en algoritme i et programmeringssprog, er ofte en tilfredsstillende opgave. Fordelen ved at arbejde med at få en figur eller funktion tegnet ud på plotterpapiret er, at man derved kan få et fysisk udtryk, en konkret fremstilling af iøvrigt abstrakte matematiske størrelser. I det følgende er vist en række vigtige procedurer, man kan få brug for under arbejdet med at udføre tegninger på plotteren. Det drejer sig om:

- 1) proceduren ELLIPSE
- 2) proceduren PARABEL
- 3) proceduren CIRKEL, CIRKELBUE
- 4) funktionen ARCUS-SINUS
- 5) funktionen ARCUS-COSINUS

Proceduren ELLIPSE fremstiller en ellipse ved anvendelse af den såkaldt "excentriske anomali".

```

4740 //*****
4750 //**ELLIPSE, ELLIPSE-BUE.*
4760 //*****
4770 //
4780 //VALG AF: CENTRUM(A,B)/FØRSTE-AKSE FØ/ANDEN-AKSE AN/EXCENTRISK ANOMALI
      BEGYNDELSESVINKEL VB/SLUTVINKEL VS.
4790 PROC ELLIPSE(A, B, FØ, AN, VB, VS)
4800   PI:=4*ATN(1)
4810   X:=A+FØ/2*COS(PI/180*VB)
4820   Y:=B+AN/2*SIN(PI/180*VB)
4830   EXEC MOVE(X,Y)
4840   FOR V:=VB TO VS DO
4850     X:=A+FØ/2*COS(PI/180*V)
4860     Y:=B+AN/2*SIN(PI/180*V)
4870     EXEC DRAW(X,Y)
4880   NEXT V
4890   EXEC MOVE_REL(0,0)
4900 ENDPROC ELLIPSE
4910 //

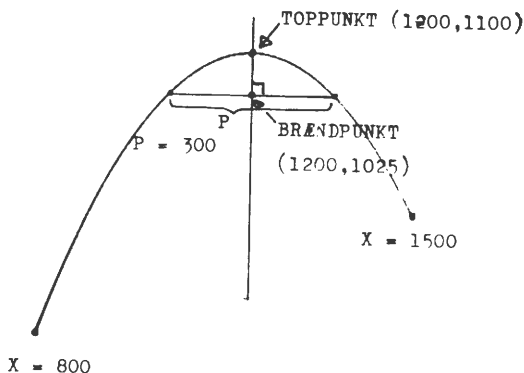
```

Proceduren Parabel fremstiller en parabel ved at anvende brændpunkt, ledelinje og parameter. Afstanden mellem toppunkt og brændpunkt er udtrykt ved  $P/4$ , hvor  $P$  er parameteren. Korden gennem brændpunktet vinkelret på parablens akse er parameteren  $P$ . Se figuren!

```

4910 //
4920 //*****
4930 //PARABEL, PARABEL-BUE.*
4940 //*****
4950 //
4960 //VALG AF: RETNING A=1 ABNING TIL HØJRE, A=2 ABNING OP, A=3 ABNING TIL
STRE, A=4 ABNING NED. /TOPPUNKT(A,B)/PARAMETER P.
4970 //STARTARGUMENT VB/SLUTARGUMENT VS/parameter fremstilling med TOPPUNKT
(0,0)  $Y^2=P*X$  hhv.  $X^2=P*Y$ . Afstand BRÆNDPUNKT-TOPPUNKT = P/4.
4980 PROC PARABEL(A, A, B, P, VB, VS)
4990 CASE A OF
5000 WHEN 1, 3
5010     VB:=VB-B
5020     VS:=VS-B
5030     IF A=3 THEN P:=-P
5040     X:=A+VB^2/P
5050     Y:=B+VB
5060     EXEC MOVE(X, Y)
5070     FOR Y1:=VB TO VS DO
5080         X:=A+Y1^2/P
5090         Y:=B+Y1
5100         EXEC DRAW(X, Y)
5110     NEXT Y1
5120     EXEC MOVE_REL(0, 0)
5130 WHEN 2, 4
5140     VB:=VB-A
5150     VS:=VS-A
5160     IF A=4 THEN P:=-P
5170     X:=A+VB
5180     Y:=B+VB^2/P
5190     EXEC MOVE(X, Y)
5200     FOR X1:=VB TO VS DO
5210         X:=A+X1
5220         Y:=B+X1^2/P
5230         EXEC DRAW(X, Y)
5240     NEXT X1
5250     EXEC MOVE_REL(0, 0)
5260 OTHERWISE
5270     EXEC NEW_PEN(0)
5280     EXEC HOME
5290 ENDCASE
5300 ENDPROC PARABEL
5310 //

```



Proceduren CIRKEL,CIRKELBUE tegner en cirkel eller cirkelbue afhængig af begyndelsesvinkel og slutvinkel. Inddata til proceduren er : koordinaterne for centrum, radius, begyndelsesvinkel og slutvinkel. En cirkel fås ved at sætte begyndelsesvinklen til 0 og slutvinklen til 360.

```

4550 //
4560 //*****
4570 //*CIRKEL,CIRKEL-BUE.*
4580 //*****
4590 //
4600 //VALG AF: CENTRUM(A,B)/RADIUS R/BEGYNDELSSESVINKEL VB/SLUTVINKEL VS.
4610 PROC CIRKEL(A, B, R, VB, VS)
4620   PI:=4*ATN(1)
4630   X:=A+R*COS(PI/180*VB)
4640   Y:=B+R*SIN(PI/180*VB)
4650   EXEC MOVE(X,Y)
4660   FOR V:=VB TO VS DO
4670     X:=A+R*COS(PI/180*V)
4680     Y:=B+R*SIN(PI/180*V)
4690     EXEC DRAW(X,Y)
4700   NEXT V
4710   EXEC MOVE_REL(0,0)
4720 ENDPROC CIRKEL
4730 //

```

Endelig vises de to vigtige funktioner arcus-sinus og arcus-cosinus, de anvendte funktioner for sinus og cosinus. Disse to funktioner er ikke til rådighed i standard Comal-80, men da de ofte finder anvendelse, bringes de her.



```

5310 //
5320 //*****
5330 //*ARCUS-SINUS.*
5340 //*****
5350 //
5360 FUNC INVS(E)
5370   PI:=4*ATN(1)
5380   IF E=1 THEN
5390     RETURN 90
5400   ELIF E=(-1) THEN
5410     RETURN -90
5420   ELSE
5430     RETURN ATN(E/SQR(1-E*E))*180/PI
5440   ENDIF
5450 ENDFUNC INVS
5460 //
5470 //*****
5480 //*ARCUS-COSINUS.*
5490 //*****
5500 //
5510 FUNC INVC(O)
5520   PI:=4*ATN(1)
5530   IF O=1 THEN
5540     RETURN 0
5550   ELIF O=(-1) THEN
5560     RETURN 180
5570   ELSE
5580     RETURN (PI/2-ATN(O/SQR(1-O*O)))*180/PI
5590   ENDIF
5600 ENDFUNC INVC

```

Det følgende program viser, hvorledes en ellipse kan konstrueres ved anvendelse af excentrisk anomali. Det skal bemærkes, at metoden også kan anvendes, såfremt man ønsker at konstruere en ellipse med passer, lineal og vinkelmåler. Ellipsens excentriske anomali er her 30 grader. Se efterfølgende figur!

```

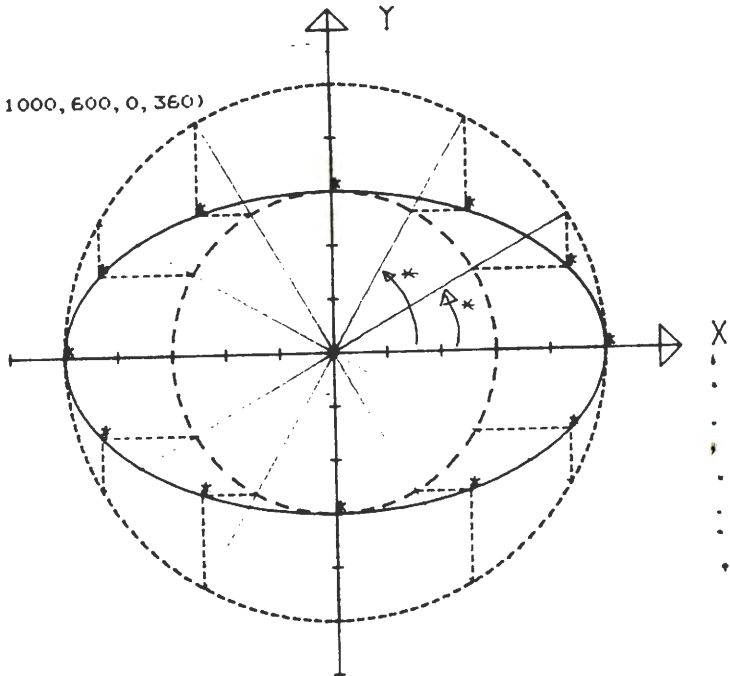
0100 PI:=4*ATN(1)
0110 EXEC CLEAR_PL
0120 EXEC MOVE(100,700)
0130 EXEC AXIS(1,100,12)
0140 EXEC ALPHA_ROTATE(2700)
0150 EXEC ALPHA_SCALE(19)
0160 EXEC MARKE(10)
0170 EXEC MOVE_REL(100,0)
0180 EXEC ALPHA_SCALE(5)
0190 EXEC ALPHA_ROTATE(0)
0200 EXEC PRINT("X")
0210 EXEC MOVE(700,100)
0220 EXEC AXIS(0,100,12)
0230 EXEC ALPHA_SCALE(19)
0240 EXEC MARKE(10)
0250 EXEC MOVE_REL(100,0)
0260 EXEC ALPHA_SCALE(5)
0270 EXEC PRINT("Y")
0280 EXEC LINE_TYPE(1)
0290 EXEC LINE_SCALE(10)
0300 EXEC NEW_PEN(2)
0310 EXEC CIRKEL(700,700,500,0,360)
0320 EXEC LINE_SCALE(20)
0330 EXEC CIRKEL(700,700,300,0,360)
0340 EXEC LINE_TYPE(0)
0350 EXEC NEW_PEN(3)
0360 FOR V:=30 TO 360 STEP 30 DO
0370   EXEC MOVE(700,700)
0380   EXEC DRAW_REL(500*COS(PI/180*V),500*SIN(PI/180*V))
0390   EXEC LINE_TYPE(1)
0400   EXEC LINE_SCALE(10)
0410   EXEC MOVE(700+300*COS(PI/180*V),700+300*SIN(PI/180*V))
0420   EXEC DRAW(700+500*COS(PI/180*V),700+300*SIN(PI/180*V))
0430   EXEC MOVE(700+500*COS(PI/180*V),700+500*SIN(PI/180*V))
0440   EXEC DRAW(700+500*COS(PI/180*V),700+300*SIN(PI/180*V))
0450   EXEC ALPHA_SCALE(3)
0460   EXEC PRINT("*")
0470   EXEC LINE_TYPE(0)
0480 NEXT V
0490 EXEC NEW_PEN(4)
0500 EXEC ELLIPSE(700,700,1000,600,0,360)
0510 EXEC NEW_PEN(0)
0520 EXEC HOME

```

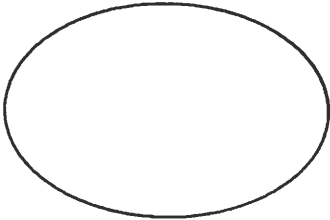
ELLIPSENS EXCENTRISKE ANOMALI.

Dette program vil i forbindelse med  
PROCEDURER for PLOTTEREN vise hvorledes  
de enkelte punkter på ELLIPSEN fremkom-  
mer.

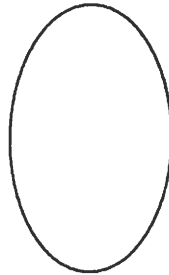
\* VINKLEN = EXCENTRISK ANOMALI.



Følgende korte program kan udtegne ellipser og parabler ved brug af de tidligere angivne procedurer.



LINIE 0160



LINIE 0170

```

0100 EXEC CLEAR_PL
0110 EXEC DRAW_REL (1700, 0)
0120 EXEC DRAW_REL (0, 2700)
0130 EXEC DRAW_REL (-1700, 0)
0140 EXEC DRAW_REL (0, -2700)
0150 EXEC NEW_PEN (2)
0160 EXEC ELLIPSE (400, 2400, 600, 400, 0, 360)
0170 EXEC ELLIPSE (1000, 2000, 300, 500, 0, 360)
0180 EXEC ELLIPSE (500, 1000, 450, 300, 100, 300)
0190 EXEC NEW_PEN (3)
0200 EXEC PARABEL (4, 1200, 1100, 300, 800, 1500)
0210 EXEC NEW_PEN (0)
0220 EXEC HOME

```

Det bemærkes, at der ved konstruktion af ovenstående procedurer og programmer er anvendt en række procedurer som en slags hjælpeværktøjer til styring af plotteren. Man vil kunne få det samlede sæt af procedurer og programmer tilsendt ved henvendelse til redaktionen.

## EDB I UNDERVISNINGEN

Karsten Juul-Olsen, TICA

---

Der tales og debatteres i disse år meget om undervisning - I, - OM, og - MED edb.

Ikke mindst i forbindelse med indførelsen af datalære i folkeskolen og gymnasiet har diskussionen fået ny næring. På landets tekniske skoler har vi undervist i datalære gennem de sidste 10 år. Vi har herigennem erhvervet nogle pædagogiske og fagdidaktiske erfaringer, som vi mener, andre også kan få glæde af. Det kræver imidlertid, at vi selv får gang i en erfaringsudveksling. Denne udveksling kan virke som katalysator for en proces, hvor pædagogisk nytænkning udvikles, så de undervisningspotentialer, der ligger i datamaskinerne kan blive udnyttet bedre.

I denne artikel har jeg forsøgt at indkredse nogle af de generelle erfaringer, som er gjort i forbindelse med datalæren på basisåret. Vi vil ved en senere lejlighed forsøge at beskrive konkrete undervisningsforløb. Således vil der i et af de næste numre blive rapporteret om undervisning MED edb, i form af et Comus-materiale omkring informationssamfundet.

Edb-undervisningen på efg-niveau har hovedsageligt drejet sig om undervisning I edb. Helt konkret har det overvejende drejet sig om programmering i Comal-80. Datamaskinerne har stået i centrum, og eleverne har haft fingrene på tastaturet. Det induktive princip har været fremherskende. De mest primitive opgaveformuleringer har været af typen: indskriv dette program og prøvekør det - hvad sker der? Efterhånden har vi fået et større repertoire og kan i dag efter drøje pionerår, mønstre

en undervisning som, når den er bedst, opfylder de ret ambitiøse læreplaner.

## Undervisning i edb

Programmeringsorienteret undervisning risikerer at blive hængende i syntaktiske spidsfindigheder, hvor udbyttet i første omgang blot bliver beherskelse af programmeringssproget og forhåbentlig en fornemmelse for datadisciplin (- hæsligt ord!). I bedste fald får eleverne tillige en fornemmelse af, hvilken slags redskab en datamaskine er, og hvilke konsekvenser brugen af edb vil få. Det kan i sig selv være godt nok, men det virkelige udbytte af programmeringsorienteret undervisning ligger nok i muligheden for at lære eleverne en generel problemløsnings-teknik. Det kræver, at undervisningen fokuserer på områderne: problemformulering, analyse, algoritmisering og kontrol. Lærerne må altså tvinge eleverne til at bearbejde problemerne ved hjælp af papir og blyant. Vi har haft gode erfaringer med at gennemgå hele arbejdsgangen i fællesskab.

### 1. Opstil problemformulering

Man har et problem, hvis man befinder sig i en situation, hvor noget fungerer uhensigtsmæssigt, uoverskueligt eller uforklarligt. For eleverne skal det konkretiseres. Man kan tage udgangspunkt i en situation, de fleste kender til. Case: En avis har ansat 200 avisbudere og skal hver uge lave en lønudregning for budene. Det er møjsommeligt at udregne i hovedet. Avisen overvejer derfor at benytte en mere hensigtsmæssig metode. Man kan nu sammen med eleverne afgrænse den nøjagtige problemformulering, og diskutere andre lignende problemer.

### 2. Analyser forskellige løsninger

Man kan prøve at starte med den enkleste løsning og afveje fordele og ulemper ved at løse problemet ved elektronisk data-

behandling. Senere kan udvidelser skitseres og diskuteres. Man analyserer, hvilke oplysninger der skal foreligge, og hvordan de skal indsamles.

### 3. Formuler en algoritme

Vi bruger en "pseudokode" til at beskrive algoritmer. Eleverne bliver relativt let fortrolige med denne metode. "Pseudokoden", som viser hovedfunktionerne i et program, er med til at sikre, at programmerne bliver velstrukturerede. I øvrigt er netop algoritmeformuleringen noget som bør trænes og øves så meget som muligt. Algoritmeformuleringen kunne for ovenstående simple eksempel være:

```

INDLÆS løn pr. avis
BEREGN bruttoløn
UDSKRIV bruttoløn
  
```

Senere udvidelser kan så kræve brug af selektion og iteration (valg og gentagelse). Det vigtigste er, at man i starten vælger simple problemstillinger, som er genkendelige for eleverne. Senere kan man i fællesskab behandle mere komplekse størrelser.

Selve arbejdet med at overføre den endelige algoritme til programmeringssproget må ikke blive PROBLEMET. For at styre uden om de syntaktiske faldgruber kan man vælge at benytte programmeringssprog, som er meget tolerante overfor syntaksfejl. Det kunne være myresnak eller logo, men måske mere perspektivrigt: programmering i Comal-80, evt. ved hjælp af procedurer, som læreren har fremstillet.

Benyttes en sådan velstruktureret arbejdsmetode vil man opleve, at hovedvægten i arbejdet ret hurtigt skubbes over på det reelle indhold i problemområdet. Det bliver altså indholdsmæs-

sige spørgsmål af karakteren: hvordan regner man egentlig skatten ud?, og struktureringen af "virkelighedens" problemområder, -algoritmebygningen-, som optager sindene, - mere end det at omsætte de fremstillede algoritmer til programsprog.

#### 4. Kontroller løsningen

Et af de meget væsentlige aspekter ved problemløsning ved hjælp af edb er fejlfinding. En rationel fejlfindings-teknik er en kvalifikation, som vil blive efterspurgt fremover. Men også i mange andre, også private, sammenhænge vil en logisk fremgangsmåde i fejlsituationer have værdi. Logisk fejlfinding efter udelukkelsesmetoden er hensigtsmæssig i næsten enhver fejlsituation. Læreren skal derfor i fejlsituationer ikke bare være "trollmanden", som med et snuptag ordner situationen. Eleverne skal stimuleres til selv at tænke situationen, algoritmen, igennem, og opstille løsningsforslag. En "skrivebords-test" vil være uundværlig i disse sammenhænge.

Man må altså indarbejde en arbejdsform, som lægger vægt på:

- problemafgrænsning
- problemformulering
- analyserende spørgsmål
- vurdering
- formulering af algoritmer
- efterkontrol

Det vil fremme indlæringen af nogle generelle kvalifikationer, som også kan finde anvendelse, hvor det ikke netop drejer sig om edb-programmer. Gennemføres programmeringsundervisningen efter disse principper, hvor de almene begreber sættes i højsædet, vil der kunne finde en relevant indlæring sted.



## Undervisning MED edb

Undervisning MED edb omtales ofte som datamatformidlet undervisning (DFU); men der findes flere andre betegnelser. Ved denne undervisningsform er datamaskinen medie og værktøj i undervisningen. Man lærer altså ved hjælp af datamaskinen nogle færdigheder og sammenhænge i forskellige fag.

Udviklingen af pædagogikken i forbindelse med denne dataformidlede undervisning er endnu kun i sin vorden. Generelt kan man sige, at der alt for ofte præsenteres "stenalderpædagogik på nye maskiner". Udbuddet af undervisningsprogrammer er ikke overvældende stort. Læreren vil derfor ofte stå i en situation, hvor det vil være påkrævet selv at kunne fremstille undervisningsmaterialet. Det er ikke så stort et problem, hvis læreren tilfældigvis også er en habil programmør. I så fald vil læreren kunne anvende et af de generelle programmeringssprog til at skrive sit program. Med de arbejdsbyrder der i forvejen hviler på lærerne, kan det måske nok knibe med tiden. Men der er altså ikke på forhånd sat snævre grænser for, hvilken type af undervisning, der kan fremstilles. Læreren står dermed i en situation, som i hvert fald formelt giver frie udfoldelsesmuligheder, såvel pædagogisk som metodisk.

Det er jo imidlertid ikke - og vil heller aldrig blive - sådan, at alle, der anvender datamatformidlet undervisning, samtidig er erfarne programmører. Derfor vil det for de fleste være mere relevant at benytte et af de forfattersystemer, som netop er udviklet for, at lærere uden programmeringskendskab alligevel kan udvikle eget programmateriale. Der findes en række af disse forfattersystemer: Plato, Tencore, Comcats (DUS), Comus m.fl. Af disse har imidlertid kun Comus og Comcats aktuel interesse, idet de er de eneste, der kan køres på de datamaskiner, som for øjeblikket anvendes indenfor skolesektoren.

Før man køber et forfattersystem, må man gøre sig klart, at man også får en pædagogisk holdning med i købet. Det behøver dog ikke være ensbetydende med, at man så er låst fast til denne pædagogiske linje. Selv om man ikke automatisk sælger sin sjæl til datamaskinpædagogikken, vil det under alle omstændigheder være fornuftigt at overveje, hvilken pædagogisk grundholdning, der ligger bag det forfattersprog/program, man vælger at benytte. Der findes groft sagt 3 grundholdninger i pædagogikken, som er specielt relevante i forhold til DFU.

1. Den materialistiske pædagogik, som er meget dominerende repræsenteret i de eksisterende undervisningsprogrammer. Grundantagelsen er, at det enkelte individ er et produkt af summen af de stimuli, det har været/er udsat for. Opgaven bliver dermed at tilrettelægge stimuli, så individet udvikler sig i den ønskede retning. En repræsentant for denne pædagogik er den amerikanske psykolog og adfærdsforsker Skinner, men også i Østeuropa er denne strukturerede pædagogik fremherskende. Denne grundholdning har udmøntet sig i programmerede undervisningsforløb, også før datamaskinen blev taget i brug. Selvkontrollerende prøver, multiple-choice opgaver har deres ideologiske udspring i disse teorier. Edb-programmerne indenfor denne tradition er karakteriseret ved en adfærdsorienteret indlæringsstrategi. Hvis eleven besvarer de stillede spørgsmål korrekt, dvs. udviser den ønskede adfærd, udskriver maskinen den programmerede ros - tæller evt. points op - og går videre til næste spørgsmål. Der findes kun eet rigtigt svar på hvert spørgsmål - nemlig det svar, som konstruktøren af programmet har bestemt er det rigtige. Analyse og diskussion hører ikke rigtig hjemme her.

2. Pædagogikkens rolle er i følge den idealistiske tradition at skabe mulighed for, at individet kan realisere sig selv fuldt ud. Dette kan bl.a. ske ved at lade eleverne fortolke og

forholde sig til omverdenen. Simulationsprogrammer er eksempler på edb-programmer, der svarer til denne grundholdning. Mange naturvidenskabelige fag arbejder med en modelbygning af virkeligheden.

Under nogle simplificerende antagelser (alt andet lige) kan man opstille beregningsmodeller, således at konsekvenserne kan udregnes. Vi kan f.eks. tænke på en model til beregning af varmetabet i et parcelhus. Eleverne kan afprøve konsekvenserne af at isolere med yderligere 100 mm mineraluld på loftet, udføre hulmursisolering, isætte 3 lags termoruder osv. Gennem en sådan modelbearbejdning forøges elevens forståelse af både den programmerede model, og (forhåbentlig) den bagvedliggende virkelighed. Med et simulationsprogram kan man altså efterligne den virkelighed, som man ønsker at lære eleven noget om. Når visse handlinger indtastes, simulerer programmet de reaktioner, som ville have fundet sted i "virkeligheden", og konsekvenserne af handlingerne indlæres dermed. Der ligger selvfølgelig også i denne forbindelse vigtige diskussioner omkring, hvilke udtalte præmisser der bygges på i modelsammenhængene. Det gælder ikke mindst, når det drejer sig om værdiladede emner som f.eks. simuleringsprogrammer over den økonomiske udvikling i samfundet eller over udviklingen i fiskebestanden i Nordsøen.

3. Endelig, og som en slags syntese af de foregående, ser den dialektiske pædagogik forholdet mellem menneske og omverdenen som et samspil. Mennesket påvirker gennem sine handlinger omverdenen, men påvirkes samtidig også selv heraf. Som eksempel kan man tænke på udviklingen af datamaskinerne. Opfindelsen af 1. generations datamaterne, ændrede på produktionsforholdene, hvilket bl.a. muliggjorde frembringelsen af 2. generations datamaskiner. Udviklingen af 2. generations datamaskiner ændrede på produktionsforholdene, hvilket osv. osv...

Edb-programmer, som lever fuldt op til denne dialektiske grundholdning, er ikke sådan lige at stampe op af jorden. Logo eller myresnak er eksempler, der på det logiske, relationsgeometriske område kan udmønte denne pædagogik. Comus er, som vi skal se, et af de bedre bud på et forfattersystem, der kan benyttes i en undervisning med en dialektisk pædagogik.

### Forfattersystemer

Her skal kun omtales COMUS og DUS (comcats), som begge er dansk udviklede og relevante i forhold til skole-datamaterne. COMUS er et af de mere interessante bud på et forfattersystem, som lægger vægt på, at det er eleverne, der styrer indlæringsforløbet. COMUS udnytter samtidig de muligheder for hurtig informationssøgning, der nok er datamaternes største force i forbindelse med undervisning. I øvrigt er COMUS skrevet i Pascal, hvilket principielt gør det muligt at få systemet implementeret også på andre skoledatamater end ICL's Cometer. COMUS er bygget op omkring 4 editorer, dvs. redigeringsprogrammer, hvor tekst eller billeder kan indskrives. Eleverne kan hente opgaveformuleringerne i en opgaveeditor. Opgaverne kan være af 2 forskellige typer:

1. Opgaver som kræver et langt åbent svar. Eleverne løser opgaven og skriver deres løsning/rapport ind på COMUS. Forfatteren kan i forvejen ved hjælp af stikord antyde en mulig løsningstekst. Eleverne kan nu analysere og sammenligne deres egne og forfatterens løsningsforslag. Der findes ikke kun en rigtig løsning.

2. Opgaver som kræver et kort kontant svar. Eleven kan indtaste sit svar i et svarfelt samt fremhente forfatterens svar i et løsningsfelt. Elevens svar evalueres ikke af COMUS, og det er derfor op til eleven selv at sammenligne med det svar, forfatteren har angivet som det rigtige.

Fælles for opgaverne er, at eleverne får informationer til besvarelsener ved at "bladere" rundt i en række informationsskærm-billeder med tekst eller grafiske fremstillinger, som forfatteren har indlagt. Eleverne kan selv afgøre, hvilke oplysninger der er relevante i forbindelse med den konkrete opgave.

DUS er i sit grundmønster præget af adfærdsorienteret pædagogik. Ved hjælp af en forfatter/lærer-del indlægges forskellige spørgsmål - svar - samt reaktioner ved forkerte eller rigtige svar. Opbygning af et undervisningsforløb kræver, at stofområdet kan afgrænses og struktureres temmelig præcist. Når eleverne svarer på de stillede opgaver, evalueres svaret, og næste spørgsmålsniveau kan være afhængigt af resultatet af tidligere besvarelsener. De dygtige elever kan få ekstra opgaver på højere niveau, mens de svagere elever får mere vejledning og lettere opgaver. DUS-folkene mener selv, at anvendelsen af systemet kan udjævne forskellene i vidensgrundlag i en klasse, og dermed danne grundlag for en mere udbredt fælles klasseundervisning. DUS-systemet kan kobles til f.eks. dias og videoapparater og dermed styre disse medier i et undervisningsforløb.

Uanset hvilket system der anvendes, kræver det, at man som lærer gennemtænker og beskriver sit materiale meget grundigt. Erfaringerne viser, at det tager ret lang tid at konstruere noget, der er brugbart. Det er medvirkende til, at der endnu ikke findes ret meget færdigt materiale. Det er et åbent spørgsmål, om det bliver en overskuelig opgave for den enkelte lærer at konstruere sine egne programmer. Sandsynligvis vil udviklingen vise, at området bliver en parallel til markedet for trykte undervisningsmidler.

## Undervisning OM edb

Denne del af undervisningen sigter mod at anskueliggøre de konsekvenser, som anvendelsen af edb og edb-systemer kan få i samfundet. Det er et meget omfattende emne, fordi edb griber ind på så mange samfundsmæssige områder.

Vi har gode erfaringer med at gennemføre undervisningen i et forløb, som omfatter et eller flere virksomhedsbesøg. På virksomhedsbesøgene kan eleverne se et edb-system i drift, få forklaret arbejdsgangen i forbindelse med edb og evt. snakke med medarbejderne om forholdene før og efter indførelsen af edb, m.v.

Besøgene skal selvfølgelig være velforberedte. Det kan man sikre bl.a. ved at lade eleverne arbejde med et mindre program, som simulerer det professionelle. Det kan være et lagerstyringsprogram, et kasseterminalsystem, et banksystem, offentlige registre eller maskinstyringsprogrammer. Gennem arbejdet med et lille system kan eleverne få en fornemmelse af, hvilke slags opgaver, der kan klares med edb, og hvilke ændringer i arbejdsforhold og arbejdsindhold det medfører. Når virksomhedsbesøgene er gennemført, kan man evt. vende tilbage til simulationsprogrammerne, ændre i dem og herunder diskutere, hvilke betingelser der skal være opfyldt, for at edb kan indføres på en acceptabel måde.

Som sagt er emnet meget omfattende og aktuelt materiale vil altid kunne skaffes til veje i pressen.

## KRAV TIL PROGRAMMEL

Karsten Juul-Olsen, TICA

---

Efterhånden som der bliver større udbud af færdige programmer, samtidig med at flere og flere bliver i stand til selv at lave programmer, stiger behovet for at kunne vurdere programmerne på baggrund af nogle relevante bedømmelseskriterier. Der skal være mulighed for hurtigt at bedømme, om et givet program holder en tilstrækkelig god standard med hensyn til sikkerhed og brugerdocumentation m.v. Det vigtigste overordnede begreb er, at brugeren i størst muligt omfang kan tilpasse systemets virkemåde til sin situation. Dvs., at programmet er tolerant, og at det giver valgmuligheder.

Jeg har her prøvet at samle en række kriterier, som jeg mener er rimelige at overveje i forhold til de fleste typer af programmer, og som i hvert fald er relevante, når der er tale om programmer til datamatformidlet undervisning (DFU). Denne liste opstiller idealkrav, og ikke alle kan forventes opfyldt i ethvert program.

### 1. Systemet skal være åbent

Dermed menes, at systemet ikke må være for fastlåst på helt specifikke indlæringsmål. Hvis man i længden skal have noget udbytte af programmet, må det være muligt at bruge programmet i forskellige undervisningssituationer med forskellige indlæringsmål.

## 2. Programmet skal være veldokumenteret

Hvis man blot en gang har forsøgt at starte et program med 'Run', for derpå at få en blank skærm, et spørgsmålstegn og en blinkende markør at se, så vil man vide at værdsætte ordentlig vejledning eller brugerdokumentation. Denne brugerdokumentation blev tidligere hovedsagelig varetaget af pæne mænd i grå habitter, som tog rundt med deres programmer og forklarede og instruerede i brugen af dem. Nu om stunder er dokumentationen ofte indbygget i programmerne. Det er således ikke ualmindeligt, at 75% af et kørende program opfyldes af brugergrænsefladen. Grænsefladen er forklarende tekster o.lign., som systemet sender til brugeren i de forskellige sekvenser af programmet. I denne situation er man undertiden havnet i den anden grøft med hensyn til brugerdokumentation. Man starter programmet, og frem vælter skærbillede efter skærbillede af forklarende tekst. Ofte er teksten styret af programindbyggede forsinkeløkker, så man enten dårligt nok når at læse teksten, eller alternativt sidder og trommer fingre, medens en eller anden imaginær dårlig læser får snøvlet sig færdig. Hvis et program kræver mere end højst et skærbillede instruerende tekst, bør den foreligge på papir, og kun vises frem på skærmen efter bestilling. Intet er mere irriterende end at skulle trækkes igennem den samme forklaring den 10.ende gang, programmet benyttes. Endelig skal den forklarende tekst stå på skærmen, så længe som brugeren finder det nødvendigt, dvs., at brugeren f.eks. ved at taste 'Return' kan komme videre i programmet.

## 3. Programmet skal være differentieret

Det vil sige, at der i programmet er taget højde for individuelle forudsætninger hos brugerne. Det kræver, at programmet kan afvikles på forskellige niveauer, både hvad angår sværhed og udstrækning/omfang. Der skal i denne sammenhæng også være mulighed for at indstille evt. tærksler for hjælp ved fejlsvær. I et traditionelt matematikprogram vil det f.eks. sige, at



eleven/læreren skal kunne bestemme regnearten, regneenhedens størrelse, antal stykker og antal fejlsvar, før det rigtige svar præsenteres.

#### 4. Programmet skal være interaktivt

Alle programmer er stort set interaktive på ordreniveau. Dvs., at man kan skrive 'Run' eller 'Escape' eller 'Delete' osv. til dem. Der er større variationer i interaktiviteten på kommunikationsniveau, dvs. muligheden for at indlæse - spørge - vælge, når programmet kører. Forskellene opleves som forskelle i brugerfladen, og dermed i grænsefladen mellem bruger og programmel. Man kan skelne mellem forskellige strukturer i brugerfladen:

##### a. Kommando-orienteret

Grafisk fremstillet ville strukturen minde om en stjerne, hvor brugeren står i centrum. Med kommandoer (evt. funktionstaster) kan brugeren aktivere mange forskellige underprocedurer. Et eksempel på denne kommunikationsform er teksteditorer, hvor forskellige redigeringsfaciliteter bestilles med en enkelt kommando. Fordele ved denne brugerflade er, at man hurtigt bevæger sig ud til den ønskede programfunktion. Denne form for interaktivitet stiller imidlertid ret store krav til forhåndsviden hos brugeren. Brugeren skal have overblik over de forskellige muligheder i kommandosættet, og over hvordan man bevæger sig rundt i systemet. Minimumskrav til denne brugerflade er, at man altid kan hente en hjælpetekst frem på skærmen. Ofte må man kræve en supplerende skriftlig brugerdokumentation til denne kommunikationsform.

##### b. Menu-orienteret

Denne brugerflade rummer en mere fastlagt kommunikationsform. Brugeren præsenteres for en række muligheder på

skærmen. Et valg af en af disse muligheder bringer evt. brugeren ned til en undermenu, hvor der igen præsenteres en række muligheder. Sådan fortsættes, indtil brugeren når til udførelsen af den ønskede procedure. Herefter returneres til hovedmenuen. Kravene til brugerforudsætninger er ikke helt så vidtgående i denne brugerflade. Til gengæld er kommunikationsformen mere fastlåst i sin struktur og ikke så hurtig. Den øvede bruger kan opfatte menustyringen som et snærende bånd. Brugeren kan ikke umiddelbart selv styre kommunikationen, men dette sikrer på den anden side en mere fejlsikker betjening af programmet. Minimumskrav til en sådan brugerflade er, at der fra hver undermenu er mulighed for at returnere til den foregående menu. Ved menustrukturer med mange lag af undermenuer kan det være ønskeligt, at der er mulighed for at hente en strukturoversigt. Brugerdokumentation skal ikke blot stå i starten af programmet, men må være fordelt på de enkelte menuniveauer.

#### c. Valgorienteret

Denne kommunikationsstruktur leder brugeren gennem en række valg, som der kan svares ja eller nej til. Denne selektion fører sluttelig frem til den ønskede procedure. Brugeren behøver således ingen forhåndskendskaber. Programmet kan være meget sikkert, idet alle fejlkilder kan være forudset af programkonstruktøren. Det er langsomt at betjene sig af og undertiden irriterende, især når man kender programmulighederne og kunne ønske sig en prompt-styring. Minimumskravet må være, at man kan fortryde et valg og returnere til foregående valgsituation. Brugerdokumentationen ligger implicit i formuleringen af spørgsmålene.

Hvilken af disse kommunikationsstrukturer, man skal prioritere højest ved valg af programmer, afhænger naturligvis af den forventede brugersituation.

Endelig vil det være formålstjenligt, at interaktiviteten også omfatter en metakommunikation. Dvs., at programmet hele tiden undersøger det indtastede og kan give meddelelser som f.eks.: "Hov! du tastede vist forkert, - Her må kun indlæses tal, - Du må kun vælge mellem de nævnte muligheder - prøv igen".

#### 5. Programmet skal kunne modificeres

For at dette krav kan opfyldes på en rimelig måde, må det være muligt at få kildeteksten at se (programlinjerne "listed" frem). Dette er et nødvendigt, men ikke tilstrækkeligt krav. En reel mulighed for at rette og ændre i programmet fås kun, hvis programmet er velstruktureret og modulopbygget. De enkelte moduler - procedurer - skulle i så fald forholdsvis let kunne lokaliseres og modificeres. Dette kræver dog et vist elementært programmeringskendskab hos lærerne.

Ovenstående krav er som sagt idealkrav, men kan tjene som en slags målestok ved vurdering af egne og andres programmer.

## CP/M-SPALTEN

Om submitfiler

v. Karsten Juul-Olsen, TICA

Hjælpeprogrammet SUBMIT.COM har nogle faciliteter, som kan være ganske praktiske at benytte sig af. Submit-rutinen giver nemlig mulighed for at få udført CP/M-kommandoer "automatisk" efter hinanden. Det kræves blot, at der laves en sub-type fil, hvori kommandoerne lægges linje for linje. Dette foregår nemmest ved brug af en teksteditor, f.eks. Bogikas editor eller CP/M's egen ED.COM. Sidstnævnte er dog noget besværlig at arbejde med, da det kun er en linjeorienteret editor. I sub-filen kan afgives alle de kommandoer, som normalt kan benyttes på CP/M-niveau, dvs. i den syntaks, som er anvendelig ud for A

Det er for det første alle de indbyggede kommandoer: TYPE, ERA, DIR, REN, SAVE.

Men også alle de såkaldte transiente standard kommandoer som: SYSGEN, PIP, ED, STAT, ASM, LOAD, DUMP, DDT, SUBMIT og MOVCPM. Kommandolinjen kan endvidere bestå af kald af en COM.-fil, f.eks. COMAL-80.

Når linjerne er indskrevet ved hjælp af tekst-editoren, eksempelvis således:

```
DIR
STAT *.*
COMAL-80
```

gemmes de på disketten med typeangivelsen SUB, f.eks. TEST.SUB

Nu indlæses CP/M-operativsystemet, og de indskrevne kommandoer eksekveres ved at skrive ud for A

SUBMIT TEST

Under udførelsen af kommando-linjerne kan man se de aktuelle kommandoer blive kaldt ud for A

SUBMIT-kommandoen kan være nyttig, hvis en kommandosekvens skal bruges regelmæssigt, men er også brugbar i forbindelse med automatiske opstartsrutiner - AUTOLOAD. Denne brug vil vi se nærmere på i næste nummer.

## HUMANISMEN TAGER TIL GENMÆLE

**Bent Madsen og Arne Mortensen:  
Computermagt og menneskeværd.**

Københavns Bogforlag. 1985, 132 s. kr.118

anmeldt af

Niels Brøndum Josephsen, TICA

"Det vil være en underdrivelse at konstatere, at "der går et spøgelse gennem Europa". Den slags konstateringer var patetiske for omkring 100 år siden, da Marx og Engels indledte deres "Manifest" med ordene: "Et spøgelse går gennem Europa". Spøgelset, de henviste til, var kommunismen, der netop gik gennem Europa, hvor det satte sindene i bevægelse og det hemmelige politi i alarmberedskab.

De tider er forlængst forbi. Vi er på vej mod datasamfundet, registerstaten, teknofascismen eller måske det elektroniske paradis.

Fosterteknologien, gensplejsningen og informatiktilværelsen er allerede realiteter, som præger hverdagen. Fra isranden ved Thule til de yderste forrevne skær på den chilenske side af Ildlandet huserer et nyt spøgelse. Dets navn er COMPUTER, og det adskiller sig på mange måder fra kommunismens næsten hyggelige (og i dag ganske omgængelige) spøgelse. Det nye spøgelse er ikke forfulgt af tzarens politi, det er ikke belagt med censur, det udsættes ikke for berufsverbot, og det er ikke genstand for klassekamp - slet ikke! Det udgør en sikker bestanddel af alle tanker om tiden - og da især om fremtiden - dets eksistens er på alles læber, dets trusler indgår i alle ordentlige mareridt, dets løfter præger alle drømme om snarlig bedring for patienten.

Det nye spøgelse COMPUTEREN huserer i verden. Det har fortrængt alle hidtidige spøgelse, frelserskikkelser, djævlfigurer, medicinmænd og åndemanere(...) Overalt, hvor det viser sig, bliver det omgærdet med hellig ærefrygt, krybende angst, fascination, væmmelse, forherligelse, tilbedelse, forhåbning, trosforestillinger og hallucination. Ja undertiden ligefrem religiøs liderlighed! (...)"

"Der ser vi så den europæiske civilisation, der jo ellers har næret sin humor og sit vid ved latterliggørelsen af de "primitive", "vilde" kulturer, der gik på knæ for conquistadorernes glasperler og tarvelige spejle. De fremmede kulturfolk tillagde disse teknologiske frembringelser magisk kraft. Første gang de så en europæer til hest, troede de gudhjælpemig, at mand og hest var et, og at der var tale om en ny slags Gud - hvor primitivt! Hvor latterligt, at disse mennesker gik på knæ for et par glasperler og et håndspejl, hvor sølle at de sendte deres stolte kulturer i grus for en ret linser!

Der ser vi nu denne fnisende og chauvinistiske europæiske kultur endelig selv være bragt på knæ og selv være i færd med at sælge ud af idealerne og værdiforestillingerne - ikke for en ret linser, men for en elektronisk maskine, der kan sige noget så inciterende som BIB-BIB".

Med sådanne verbale breddsider har humanismen omsider indledt en kulturpolitisk modoffensiv mod den omsiggribende, løbske teknokultur.

Bogen "Computermagt og menneskeværd" er et skarpt formuleret, sprudlende veloplagt opgør med tidens almindeligt udbredte knæfald for datamaten og datakultens selvsikre præsteskab.

Bogen er skrevet af to forfattere: Bent Madsen, der er seminarlærer og har ledet kurser om pædagogik og informatik, og Arne Mortensen, der er seminarierektor og cand.pæd., og har studeret datamatstøttet undervisning på Pennsylvania University i USA i 1978. Derudover er han ekstern lektor i pædagogik på Danmarks Lærerhøjskole og har ledet kurser i pædagogisk informatik for Undervisningsministeriet.

D'herrer gjorde sig for nylig bemærket ved at foranstalte deres egen minidemonstration foran Christiansborg, hvor 300 interesserede tilhørere var stimlet sammen til en konference med den amerikanske psykolog og computereksperter Seymour Papert, der har udviklet edb-undervisningssproget LOGO. Makkerparret Madsen og Mortensen uddelte ved den lejlighed løbesedler med paroler som: "Datamaskinen er katekismus, Grundtvig er dialog", "Nej til US-kulturimperialisme" og "LOGO ud af Danmark".



De to provokatører er godt klar over, at det midt i datamaternes tilsyneladende uafvendelige sejrsgang er helligbrøde at stable sig op og tage til orde mod teknologi og teknofix. Det er blasfemisk at anfægte en ubestridelig og almægtig sandhed. Det er at træde de tunge datadrenge over tærne. Eller at bide de fine damer i ben eller barm. Og det er lige netop det, der er deres erklærede hensigt og program, indrømmer de uden blusel. De påtager sig gerne katterollen i forsvaret for de humanistiske idealer og menneskelige interesser.

Dagligt kan man læse fascinerede avisreportager om computerteknologi, edb-anvendelser og mikroelektroniske innovationer - og dog er computeren en automat, om hvis dynamik, principper og konsekvenser de færreste er tilstrækkeligt informeret. I betragtning af en vidt udbredt fetichering og fortrængning af computeren fra den offentlige bevidsthed er det let for koncerner og "eksperter" at mystificere den. Den personificeres og markedsføres ligefrem med en sjæl. Den personlige datamat. Venlig og kompatibel. Din 3. hjernehalvdel..

At den elektroniske databearbejdning endnu ikke er rykket tilstrækkeligt ind i centrum for en bredere offentlig diskussion, hænger bl.a. sammen med, at problemer og farer ved computerindsatsen - i modsætning til f.eks. atomkraft - ikke er "synlige". Computeren kan lydløst, med ufattelig hast og præcision samle, oplagre og bearbejde myriader af data... Der er tilsyneladende ingen grænser for Edb's anvendelsesområder.

Hidtil har den offentlige debat om informationsteknologien foregået på ensidigt naturvidenskabelige, tekniske og økonomiske præmisser. Kun nogle af edb-udviklingens virkninger på arbejde og beskæftigelse har været genstand for empirisk samfundsforskning. De humanistiske positioner er trængt i defensiven i den standende debat.



Forfatterparret Madsen og Mortensen søger nu at kridte banen anderledes op: Der er for dem tale om en kulturkamp, som de ikke betragter som en kamp mellem mennesket og maskinen, men derimod som et opgør mellem to former for kulturopfattelse. Det er teknokulturens livtag med humanismen. Hverken mere eller mindre.

Selv spagfærdigt forbeholdne røster eller tilløb til kritik bliver idag på forhånd stemplet som uansvarlig naiv maskinstorm og hånligt affærdiget som irrationel data-skræk. Madsen og Mortensen prøver at trænge igennem dette mistænkelliggørende røgslør. De ønsker ikke at blive afskrevet som romantiske bagstræbere. De foregiver ikke at ville være nuancerede, men de vedkender sig ikke nogen generel og ureflekteret maskinstorm.

Som modsætning til maskinstormeri opstiller de teknofetichismen, hvorved de forstår "den ukritiske, konsekvente og uforbeholdne tilslutning til teknologi og maskindyrkelse under enhver form og til ethvert muligt formål". Stillet over for dette skisma finder forfatterne det ikke særligt presserende at advare mod maskinstorm. For tilløb hertil er sjældne. Det er teknofetichismen derimod ikke! Teknofetichismen har som en snigende sot bredt sig ud i alle afkroge af samfundet i de senere år. Den er karakteriseret ved en skamløs mangel på beskedenhed, den er på mange områder amoralsk. Teknofetichismen vinder aggressivt ind på alle områder. De tider hvor teknologi, maskiner, værktøj og udstyr var noget som fortrinsvis blev brugt i produktionsværksteder, i fremstillingsprocessen samt i erhvervsvirksomheder er forbi. Teknisk udstyr, maskiner og tekniske eller teknologiske løsninger er for længst blevet hverdag overalt i den menneskelige tilværelse. Teknologien præger hospitaler og plejehjem, bistandskontorer og biblioteker, børneværelser og skoler.

Teknofetichismen "gider ikke (eller kan ikke) bruge tid på at diskutere etik eller rimelighed. Den er i den grad overbevist om sin egen ufejlbarlighed og effektivitet, og den tromler frem på alle områder med en sådan massiv sejrssikkerhed, at enhver forbeholden humanist må se sig henvist til skammekrogen eller genopdragelseslejren for dissidenter".

Med datamaten er der åbnet mulighed for at teknologisere en lang række områder, som hidtil har været varetaget af menneskelig omsorg, intuition og fornuft. Og det er netop det foruroligende, at den af magtfulde økonomiske, politiske og teknologiske kræfter er tiltænkt rollen som erstatning for mennesker og deres indbyrdes kommunikation.

Overfor teknofetichismens mangel på moralsk habitus, dens ubeskedenhed og manglende forståelse for humanistiske værdiorienteringer ønsker forfatterparret at trække nogle grænser. Det drejer sig ikke om at standse den teknologiske udvikling som sådan, men om at bremse datamaternes snigende invasion på områder, hvor de intet har at gøre: i opdragelse og undervisning, i pleje og omsorg, i overvågning og kontrol, i instruktion og kommunikation.

Madsen og Mortensen er fortalere for, at den fagre nye teknoverden stilles over for alternative værdier, livsformer, eksistensformer og kulturer. Det handler om "at få sat teknofetichismen på plads eller om muligt at bringe den til at indse, at mennesket ikke er en ting, at barndommen ikke er et beklageligt intermezzo, at demokrati og civilisation og menneskelighed ikke kan udvikles ved hjælp af maskiner, men kun ved hjælp af menneskelig udvikling og demokratisk dannelse".

Med dette udgangspunkt tager forfatterne så uforfærdet fat på at punktere nogle af de myter og falske forestillinger, der

fra forskellig side knyttes til den mikroelektroniske revolution.

Det går derudaf over stok og sten. Indimellem sætter de dog lidt for hurtigt over, hvor gærdet er for lavt. I et hasblæsende tempo oprulles og tilbagevises myterne om,

- at computeren skulle skabe mere arbejde og flere arbejdspladser;
- at den kun overtager det trivielle slavearbejde, så det mest spændende og udviklende arbejde bliver tilbage;
- at det er en neutral, universel maskine med uendelige anvendelsesmuligheder;
- at informationsteknologien skulle give adgang til flere informationer, sætte folk i forbindelse med hinanden og nedbryde isolationen mellem mennesker;
- at computeren muliggør decentralisering af magten, og at de folkelige bevægelser, græsrodderne, må inddrage computeren og bruge den som våben i frigørende øjemed.

Undervejs uddeles der drøje hug til højre og venstre. Visse af punkterne kunne nok have fortjent en mere grundigt underbygget argumentation. Man kunne godt have undværet bogens 20 afsluttende, tyndbenede teser. Det forekommer også for summarisk, når forfatterne serverer et udvandet sammenkog af den franske skribent Andre Gorz' tanker om "arbejdets afskaffelse". Gorz' forestillinger bygger på en række rapporter, som er fremkommet i de seneste år, om den nye teknologiske skønnede effekter på beskæftigelsen og livsmønsteret i de udviklede lande frem til år 2000. Det problematiske i hans perspektiv er, at han mod-

standsløst lader sig fange af disse fremtidsprofetier. For ham er år 2000, sådan som det udmales i fremtidsprognoserne, en realitet. Tanken er, at den ny teknologi kan afskaffe lønarbejdet. Intet kan stoppe den omsiggribende arbejdsløshed indenfor det nuværende system. Automatiseringen af produktionen reducerer den samfundsmæssigt nødvendige arbejdstid stadig mere. Denne sænkning af den nødvendige arbejdstid vil blive så omfattende, at den ikke kan modsvares af nogen tankelig ekspansion i produktionen. Hvis arbejdsproduktiviteten går mod uendelig - fordi robotter laver robotter, der laver biler osv. -, så vil der til sidst næsten ikke være flere lønarbejdere tilbage og dermed ingen til at efterspørge og købe varerne. Resultat: krise. Hertil kommer, at de nye teknologier vil være så billige, at kapitalejerne ikke vil kunne finde noget at investere i. Resultat: dobbeltkrise. Dermed åbnes vejen for et positivt alternativ: Kapitalen forlader scenen, og afløses af det arbejdsfri samfund. Mikroprocessorer og bio-teknologi gør det muligt at frigøre sig fra massearbejdets tvang og udvikle produktivkræfterne økologisk, brugsværdistyret. De gør det muligt at styre produktionsprocessen langt mere snildt og detaljeret og aflukke naturen rigdomme på en langt mere nansom måde.

Hokus pokus er den negative fremtidsutopi - arbejdsløshedssamfund, registerstat, økologisk sammenbrud - forvandlet til en positiv utopi. Mareridtet er blevet til en lykkedrøm. Men det er og bliver dog i sin kerne en drøm: at maskinerne vil overtage alt arbejde. Den nuværende produktionsmåde, som beherskes af værdiproduktionen, kan ikke befri sig fra de begrænsninger, som arbejdskraftens brugsværdi pålægger den. Det er og bliver kun den menneskelige arbejdskraft, som kan skabe mere værdi, end der i forvejen er indeholdt i den.

Stærkest i Madsens og Mortensens bog står afsnittene om maskinkulturens indtog i børneopdragelse, undervisning og omsorg. Her er de på hjemmebane.

Forfatterne refererer til projekter i England og USA, hvor man arbejder med mulighederne for at anvende mikrodatamaten i specialundervisning af handicappede spædbørn fra 3 måneders alderen. Spædbørnene skulle kunne lære at forstå sammenhængen mellem deres egne bevægelser og datamatens "svar". Forfatterne opridser udviklingsperspektiverne for sådanne projekter: 100 småbørn på 3 måneder, der alle får nøjagtig samme svar fra omgivelserne på en given bevægelse: højere eller lavere temperatur, sutteflaske styret af en plejerobot eller aktivering af et andet surrogat, der kan erstatte det traditionelle mor-barnsamspil. 3 måneders børn, der passer sig selv og lærer, at omverdenen er en imødekommende maskine. Forfatterne frygter for, at den forskning og de resultater og tekniske muligheder, der i øjeblikket udvikles inden for specialundervisningssektoren, og som angiveligt har handicappede børn som målgruppe, vil blive overført til arbejdet med "normale" børn, når det viser sig økonomisk forsvarligt; når de fornødne ressourcer er skaffet til veje; når forældre, sundhedsplejersker og pædagoger har givet op...

Så vidt er det endnu ikke kommet.

Men teknologibegejstringen leges ind allerede fra børnehavestadiet, og computerne har for længst holdt deres indtog i skolevæsenet. Folkeskolen investerer for tiden 1,5 mia. kr. i elektronisk udstyr. Lærere sendes i tusindtal på kurser for at de nødtørftigt kan få lært at bruge isenkrammet. Men søger en skole om et par brugte busser, der kan køre børnene ud i virkeligheden til egne erfaringer, så er kassen smækket i. Der er ikke penge til lejrskoler, biblioteker og skolehaver, eller til nye undervisningsformer og ekstra lærerkræfter. Datalære som valgfag i de større klasser er kun begyndelsen - næste trin bliver, at datalære bliver obligatorisk for alle børn. Og der presses fra flere sider på for at indføre datamater i

undervisningen i alle fag og på et tidligt tidspunkt i skoleforløbet, helst fra 1.klasse.

En af begrundelserne er angiveligt, at børnene skal lære at forholde sig kritisk til datamaten. Eleverne skal opøve evnen til at gennemskue informationsteknologiens samfundsmæssige aspekter og vurdere konsekvenserne af forskellige teknologianvendelser. Men der er langt fra et 7 årigt barns oplevelsesunivers og til datalære og teknologikritik. Og denne afstand bør respekteres. Forfatterne betragter forslaget om at indføre datamater i skolens yngste klasser som enten udtryk for mangel på forståelse af og indsigt i, hvad et 7 års barn er for en størrelse eller som et skalkeskjul for at få noget elektronisk legetøj ind i de små klaser, som kan bibringe børnene noget teknisk motivation. Hvad angår den proklamerede teknologikritik, så indskrænker den sig så godt som altid til, at der udsiges en masse ord og luftes nogle betænkeligheder, inden man så endelig indsætter disketten i drevet.

Også fra forældre side presses der på. Forældrene er bekymrede for deres børns jobudsigter, og det er en udbredt misforståelse, at hvis børn bare får godt med data i skolen, så vil de blive sikret en fremtidig plads på den aktive side i arbejdslivet.

Det kan derfor heller ikke undre, at indførelsen af edb som fag i folkeskole og gymnasium af den nye ungdom i vidt omfang opleves, som om den dermed fik fri adgang til en fremtid - der jo ellers så udsigtsløs og tillukket ud.

De store midler, der investeres i indkøb af maskiner, er ikke den eneste udgift, man får ved indførelse af datamatformidlet undervisning. Den egentlige flaskehals er og bliver programmerne. Det beklages vedvarende, at den tilgængelige software ikke er god nok. Den begrænser sig - også efter 10 års intenst



udviklingsarbejde i USA og Japan - til træningsprogrammer med tabeller, staveord, gloser osv. De datamatformidlede undervisningssystemer holder ikke niveau med den danske dannelsesstradition eller den pædagogiske og udviklingspsykologiske forskning. De foreliggende programtyper repræsenterer pædagogiske arbejdsmetoder, som er gammeldags i forhold til nutidige arbejdsformer i folkeskolen.

Man venter derfor med stigende længsel på, at den fornødne danske software skal vise sig. Men iflg. Madsen og Mortensen må de utålmodigt ventende lærere, konsulenter og skolenævn berede sig på at komme til at vente til dommedag. For datamaskinen er designet efter ja-nej/ rigtig-forkert/ sort-hvide modeller. Den er fremstillet ud fra en bestemt intention og konstrueret til at løse bestemte typer af problemer. Det er behovet for styring, kontrol, registrering, entydighed, formalisering, forudsigelighed og reduktion af det komplekse, der har skabt datamaten. Og disse behov afspejles i dens egenskaber. De er en del af selve dens logik og inderste væsen. Og - hævder de kategorisk - "der vil aldrig kunne udvikles software, som kan tilføre undervisningssituationen det, som vores kultur indtil i dag har overlevet på, nemlig at mennesker kan tænke og forstå og reflektere i området MELLEM sandt og falsk - mellem ja og nej".

Vi risikerer et drastisk pædagogisk og dannelsesfilosofisk tilbageslag, hvis de hidtidige undervisningstraditioner afløses af maskinpædagogik og humanteknologisk træning.

Det har hidtil været et afgørende opdragelsesideal, at et menneske bliver menneske ved at omgås mennesker. Det er gennem dialogen og det sociale samvær vi udvikler os til selvstændige væsener. K.E.Løgstrup sagde engang, at demokrati er "en skikkelig måde at være uenig på". Hvis nutidens rastløse, skrupforvirrede "bimmerbørn" skal udvikle demokratisk sindelag, hvis de

skal lære at diskutere og være uenige på en skikkelig facon, så må de nødvendigvis indgå i sociale sammenhænge, hvor det er muligt at være uenig og nødvendigt at træffe beslutninger i anliggender, der vedkommer dem selv. Men datamaskinen kan kun rumme en endimensional verden. Man kan ikke øve sig i at være uenig på en skikkelig måde med en datamat. Den kan ikke diskutere på en kreativ måde, ikke indgå kompromisser. Den kan ikke forstå det afvigende som andet end forkert. Det er en matematisk struktureret maskine, som ikke kan tvivle på en sag. Og uden tvivl er ingen sand dialog mulig. Men den kan erstatte den sande dialog, der indebærer mødet mellem bevidstheder. Den kan træde i stedet for et gensidigt erkendelsesarbejde. Den dialog, et barn kan føre med en datamaskine, er en "falsk" dialog. Der stilles ganske vist spørgsmål til eleven, men det er i virkeligheden kontrolspørgsmål, der skal sikre, at eleven faktisk har tilegnet sig lærerens/programmørens udsagn. Computeren kan bedømme præstationer, hvis udfald på forhånd er givet. Det rigtige svar står i facitlisten.

De to forfattere er oprørte over, at teknosamfundet fortsat hugger uhæmmet ind på børns erfaringsrum og udfoldelsesmuligheder. Når teknokrater i dag taler begejstret om de muligheder, der ligger i at få indført datamaten i børneopdragelsen som et nyt motiverende element, der har den egenskab, at børn både kan lege og samtidig "lære noget", ser forfatterne det som udtryk for en rendyrket rationalistisk drøm om at ekspropriere børns leg. Og de opfordrer til at prøve at tænke den tanke til ende, at man kunne "nyttiggøre" børns leg, så den "spildtid", de ellers øder bort, kunne udnyttes til teknologisk læring. Ja, tænk engang, hvis man kunne afskaffe barndommen, springe opvækst og personlighedsdannelse over, skriver de sarkastisk. Her trækker de en demarkationslinje: Såvel natur som kultur har nogle tærskelværdier, som man ikke ustraffet kan indsnævre og beskære i det uendelige. Her maner de til at kridte støvler-



ne og stå fast. De siger nej til datamater i børneopdragelsen. Nej til maskinernes overtagelse af dannelsesprocesser og mellem menneskelige relationer. Alternativet er teknologisk barbari.

Det er en bog, som sikkert vil provokere og irritere nogen; andre vil finde den foruroligende, atter andre medrivende. Den er under alle omstændigheder tankevækkende.

**Kaj Olesen, Nils Hesselberg  
Møller m.fl.: Datamater i den  
erhvervsrettede voksenunder-  
visning.**

Udredningsrapport til Teknologistyrelsen. Århus 1985, 140 s., 75 kr

**anmeldt af Bent Rasmussen,  
lektor ved Danmarks Forvaltnings-  
højskole.**

---

Lad det være sagt med det samme: Det er en fremragende bog! Man skal ikke lade sig narre af dens ydre fremtoning, hvor forsidebilledet med et stort, rødt og tydeligvis amerikansk æble, efterlader læseren med det indtryk, at nu skal man læse 140 tatskrevne sider om overførsel af amerikansk undervisningsteknologi til Danmark. Og forsidens ord underbygger den forhåndsindstilling: Datamater... erhvervsrettet... udredningsprojekt... teknologistyrelsen.

Men bogen er ganske anderledes. Forfatterne tager deres udgangspunkt i pædagogiske holdninger til og visioner om, hvad undervisning er, og hvordan datamater bør anvendes til undervisning. Den åbne interaktive kommunikation mellem mennesker er nødvendig i undervisningen og må derfor medtænkes i den pædagogiske strategi. Man har forsømt dette i den hidtidige udvikling inden for datamatstøttet undervisning, siger forfatterne, idet man har fokuseret på datamatens værdi som pædagogisk med/modspiller i indlærings- og træningsøjemed. Dermed har man fremmet individualiseret undervisning i lukkede kommunikationsprocesser, og en sådan form for undervisning kan man kun få begrænset succes med, i hvert fald i Danmark, mener forfatterne. Man har haft alt for lidt opmærksomhed på datamatens værdi som kommunikationsredskab. Disse to synsvinkler - datamatstøttet undervisning og datamaten som kommunikationsredskab - anskues i bogen som komplementære. Med kommunikationssiden kan kommunikationsprocesserne blive åbne og interaktive ved etablering af

datamatmøder, der kan bære de kendte og nødvendige funktioner: Klasseværelse, lærerværelse, elevråd, bibliotek, opslagstavle, frikvarter, "hvisken i timerne", lektiegrupper, planlægning og diskussion af undervisning, opgaveskrivning m.v.

Bogen har visioner om brug af informatik til undervisning, der på en og samme tid muliggør en ønskelig individualisering og en åben, interaktiv kommunikation mellem de involverede lærere og elever. Dette er, mener forfatterne, forudsætningen for succes på dette område, både pædagogisk og økonomisk. (Det bliver meget billigere at producere undervisningslektioner, når man ikke på forhånd skal forudse alt, hvad der kan ske). I bogen stilles talrige forslag til, hvordan vi i Danmark kan understøtte en ønskelig udvikling, som skitseret.

Hvis man ikke har behov for at blive inspireret på det pædagogiske felt, men alene ønsker information om datamater i undervisningen, så får man i bogen et forbilledligt overblik over begreber, programtyper, forfatterredskaber, leverandører, produkter, forsknings- og uddannelsesinstitutioner og deres projekter, større brugere, forsøg og eksperimenter....

Enhver, der har med undervisning at gøre - også uden for den erhvervsrettede voksenundervisning -, kan få udbytte og blive inspireret af dette udredningsprojekt til teknologistyrelsen.

## INDBYDELSE TIL CP/M -BRUGER-KURSUS

TICA afholder i august et 3-dages kursus, der henvender sig til brugere af Comet'er, som ønsker kendskab til faciliteterne i operativsystemet CP/M.

### Formål:

At deltagerne opnår en bred forståelse for samspillet mellem maskinel og programmel på en mikrodatamat og bliver i stand til at forstå og anvende CP/M. Endvidere giver kurset et grundlag for at forstå og programmere i ASSEMBLER.

### Deltagerforudsætninger:

At deltagerne har et rimeligt kendskab til højere programmeringssprog, f.eks. COMAL-80, har kendskab til det binære talsystem, samt har prøvet at arbejde med COMET eller lignende mikrodatamat.

### Indhold:

På kurset gennemgås

- \* maskinel og programmel
- \* CP/M faciliteter
- \* maskinsprog/ hexadecimale tal

### Sted:

Teknisk Informatik Center Aarhus  
Aarhus tekniske Skole  
Halmstadgade 6, 8200 Århus N.

Tid: 19. - 21. aug. 1985

Pris: 2300 kr. inkl. forplejning

### Tilmelding:

TICA, Aarhus tekniske Skole, Halmstadgade 6, 8200 Århus N.  
tlf.: 06 - 16 61 00. Henv.: Annalise Sørensen, lok. 2103.

### Opfølgning:

Brugerkurset følges op med et 5-dages kursus i efteråret. Her behandles emnerne ASSEMBLER-programmering, tekniske aspekter i CP/M og ændring/ tilpasning af CP/M's BIOS-del. Dette kursus afvikles omkr. 1. oktob. 1985.

