

3

9. ÅRGANG

JUNI 1985

INDHOLD

data lære

INDHOLD

- Undervisningsvejledningen
- Kurset i Nakskov
- Hurra for datalæren - men hvilken?
- Hvorfor æder kaninen ikke pølsen?
- Folkeskolefraktionens Nakskovkursus 1985
- Datalære i Grønland
- Micro-PROLOG i skolen
- Studieture
- Hvordan kursus i datalære
- Bog anmeldelser
- Efteruddannelse af lærere på Vestlolland

Udgivet af

FORENINGEN FOR DATALÆRE OG ANVENDELSE AF EDB I UNDERVISNINGEN

3-D grafik og perspektivtegninger på mikrodatamat

MONSTER®

Edb-programmet Monster er et tegneredskab til mikrodatamater, som gør det muligt at lave vellignende skitser og tegninger af rumlige forhold ved hjælp af en mikrodatamat.

Monster kan lave skærmgrafik på Piccolo, ICL Comet, Piccoline og Partner, samt IBM PC og Apple II. Desuden kan Monster med en tilkoblet plotter køre på alle gængse CP/M, CP/M-86 og MSDOS computere.

Velegnet når som helst rumlige forhold skal skildres.

Vi kan nu også levere en række tilbehørsprogrammer, som gør det muligt at lave Monstertegninger ved hjælp af en digitizer.

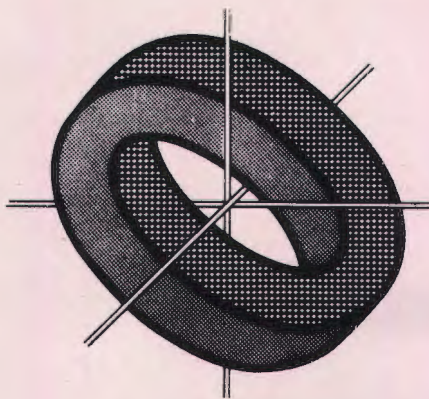
Desuden kan leveres et programmodul til tegning af solskygger på et hvilket som helst tidspunkt af året ved angivelse af breddegrad og tidspunkt.

Desuden programmer der kan lave »lagkagegrafik« og søjlediagrammer.

Anvendelsesmuligheder i matematik, fysik, formning, kemi, geografi.

Forlang brochure.

Borgens Forlag
Valbygårdsvej 33, 2500 Valby · telefon 01-46 21 00



Den nye undervisningsvejledning

Det ser ud til, at foråret sætter gang i andre safter end de traditionelle. Sidste år fik vi en lovændring og en vejledende læseplan, og nu er den nye undervisningsvejledning på trapperne. Undervisningsvejledningen er ikke endelig vedtaget til udgivelse, men vi har som faglig forening under DLF haft forslaget til udtalelse. Foreningen svarede – efter en indledning –:

- Forslaget er efter vor opfattelse tilstrækkeligt bredt formuleret i den almene del, og der gives gode eksempler på årsplaner, der med tilhørende forslag til undervisningsforløb dækker flere mulige måder at gennemføre undervisningen på. Ligeledes er de praktiske oplysninger vedrørende lokaler, holddannelse og undervisningsmidler nyttige.
- Vi finder det tilfredsstillende, at der flere steder lægges vægt på elevernes selvstændige arbejde med brug af datamaskiner, og det er foreningens opfattelse, at forslaget til undervisningsvejledning vil kunne danne et brugbart grundlag for valgfaget data-lære og den videre udvikling af beslægtede emneområder.

Som det fremgår, går den brede beskrivelse fra vejledningen igen i vort svar, men det synes klart, at vejledningen støtter den undervisning, der kombinerer praktisk brug af datamaskiner med udblik til konsekvenser for samfundslivet.

Undervisningsvejledningen vil kunne støtte os i vor undervisning i valgfaget, men det er i praksis, at indholdet skal endeligt fastlægges og udvikles. Lad os altså anvende vejledningen, udfylde den i praktisk undervisning og samtidig dermed medvirke til indholdet af den vejledning, der senest i 1990 kommer til at tegne fremtiden.

Fritz G. Knudsen.

Undervisningsvejledningen til valgfaget datalære i folkeskolen

Udvalget vedrørende datalære i folkeskolen har for nogen tid siden afleveret deres forslag til undervisningsvejledning for valgfaget datalære.

Forslaget blev sendt til høring hos en lang række parter, og Datalæreforeningens folkeskolefraktion har – som det fremgår andetsteds i bladet – sendt sin udtalelse til Danmarks Lærereforening, der så varetager det videre. **DATALÆRE** bringer i det følgende en række kommenterede klip fra forslaget til undervisningsvejledningen, der forventes at foreligge i sin endelige version i løbet af meget kort tid.

Undervisningsvejledningen indledes med formålet og i afsnittet om fagets placering citeres bl.a. nedenstående fra cirkulæret af 22. juni 1984:

»Det forventes, at folkeskolerelevant datalære-stof inden for de nærmeste år finder sin plads i skolen, således at alle eleverne får del i denne undervisning . . .«

»Man skal fra undervisningsministeriets side meget stærkt henstille til kommunerne at overveje mulighederne for at drage nytte af det apparatur, der måtte være indkøbt bl.a. med henblik på valgfagsundervisningen, ved allerede nu at lade datalære indgå i den obligatoriske undervisning . . .«

Bl.a. på denne baggrund slutter afsnittet om fagets placering med følgende signal:

Denne vejledning og den læseplan, den knytter sig til, bygger på den forudsætning, at valgfaget nødvendigvis inddrager stof, som under andre vilkår ville have været obligatorisk. Som vejledningen er udformet, er sigtet blevet bredere, end man kunne vente for valgfag. Dette har også en betydelig sammenhæng med fagets tværfaglige og almene karakter.

Det indledende kapitel afsluttes med »Generelle synspunkter«:

Undervisningen i datalære skal medvirke til at forberede eleverne til deres fremtidige tilværelse i et samfund, hvor elektronisk databehandling har en stor og stadig voksende betydning.

Denne forberedelse indeholder en orientering om, hvorledes man i det nuværende danske samfund udnytter elektronisk databehandling, og hvilke virkninger dette kan have. Yderligere indgår, at eleverne får mulighed for at tilegne sig færdigheder og udtryksformer omkring formel beskrivelse af virkeligheden, databegrebet, brug af modeller og problemløsning ved brug af datamater.

Undervisningen skal give eleverne mulighed for erkendelse af, at den tekniske udvikling også på dette område er menneskeskabt, og af, at dette giver eleverne mulighed for som borgere i et demokratisk samfund at få medindflydelse på og medansvar for, hvorledes brugen af

den elektroniske databehandling forandrer samfundets udvikling og den enkeltes tilværelse.

Undervisningen må derfor tilrettelægges, så datalære bliver oplevet som et fag, der vedkommer elevernes hverdag i og uden for skolen. Dette opnås for det første ved, at anvendelse af datamater og virkningerne heraf belyses ved brug af eksempler, der tager udgangspunkt i elevernes egen situation. For det andet opnås det ved, at undervisningen tilrettelægges, så eleverne får mulighed for oplevelse og selvvirksomhed i mange sammenhænge.

En tilrettelæggelse af undervisningen efter disse retningslinjer forudsætter, at eleverne har adgang til at arbejde med en datamat og til at udarbejde programmer til denne. Det må dog fremhæves, at det ikke er et mål for undervisningen, at eleverne opnår programmeringsfærdighed og indsigt i datamaters indretning og funktion. Dette bør inddrages i undervisningen i det omfang, det er nødvendigt for forståelsen af datamaters anvendelse.

Brugen af datamater i undervisningen skal desuden fremme elevernes muligheder for at udvikle en åben og eksperimenterende holdning til arbejdet med faget og til at erhverve viden generelt. Herigennem får eleverne desuden lejlighed til at erkende problemløsning som en proces, hvor forskellige veje fører frem til løsninger, og hvor vurderingen af en løsnings hensigtsmæssighed indgår som en vigtig del af processen.

Undervisningen skal give eleverne mulighed for at få indblik i grundlaget for den voksende brug af modelbeskrivelse af virkeligheden og den elektroniske databehandling af sådanne modeller. Blandt andet med udgangspunkt heri kan man i undervisningen belyse nogle af de økonomiske, sociale og menneskelige aspekter ved brugen af elektronisk databehandling og derved give eleverne indsigt i konsekvenserne af denne brug såvel for samfundet som for den enkelte.

Eleverne skal have mulighed for at opleve brugen af datamater som én af mange veje til at opnå viden om og indflydelse på vor omverden, at opleve brugen af datamater som en menneskelig aktivitet, hvorom naturvidenskabelige, samfundsvidenskabelige og humanistiske synspunkter mødes og brydes.

I det næste kapitel behandles undervisningens indhold, og der er et afsnit til hver af de fire hovedområder, som optræder i den vejledende læseplan. Af de indledende bemærkninger fremgår det imidlertid klart, at undervisningen ikke skal struktureres efter denne opdeling: Imidlertid må alle fire hovedområder ses i deres indbyrdes sammenhæng; fagets indhold baserer sig derfor på det samspil, der foregår i praksis mellem vor tekniske kunnen og vor kulturelle og sociale baggrund. Det vil der-

for være forkert kun at lade fagets indhold bestå af kommunikation og problemløsning eller kun af en orientering om det teknisk baserede samfund.

Det er derfor vigtigt, at indholdet i det enkelte undervisningsforløb vælges fra dette store stofområde, således at aspekter fra alle fire hovedområder inddrages.

Der er flere, der har efterlyst ordet »programmering« i den vejledende læseplan. Undervisningsvejledningen signalerer allerede i »Generelle synspunkter« hvilken rolle programmering bør indtage i datalære: arbejde med en datamat og programmering **skal indgå** som et uvæsentligt element i datalære, men det er **ikke et mål** for undervisningen, at eleverne opnår en eller anden bestemt færdighed i brug af et programmeringssprog; programmering indgår i datalære som et middel, hvilket bl.a. fremgår i afsnittet om problemløsning:

For at illustrere de to kategorier og for at give eleverne erfaring med problemløsning gennem brug af datamater skal der i forbindelse med elevernes selvstændige arbejde ved datamaterne indgå eksempler på såvel færdigudviklede programmer som programmer, eleverne selv fremstiller.

Programmering er ikke i sig selv et mål for undervisningen i datalære, men et middel til at illustrere, hvorledes datamaterne fungerer, og hvordan man arbejder med dem. Arbejdet med programmering indgår derfor som et betydningsfuldt element i problemløsningen. Der skal således indlæres nogle basale færdigheder i programmering, for at eleverne kan forstå datamatens muligheder og begrænsninger. Først herigennem skabes der mulighed for, at eleverne kan arbejde indsigtfuldt med hovedområderne i læseplanen. Arbejdet med programmering skal ikke nødvendigvis knyttes til et bestemt programmeringssprog, men skal ses som et led i en problemløsningsproces.

I afsnittet vedrørende anvendelsesområder for datamater påpeges betydningen af elevernes arbejde ved datamaten endnu engang:

Der findes mange anvendelsesområder for datamater, og man må derfor vælge en række repræsentative eksempler, der illustrerer forskellige steder, hvor datamater anvendes, og forskellige måder som datamater anvendes på.

For at eleverne kan udbygge deres forståelse af de mange forskelligartede anvendelser af edb i samfundet, er det vigtigt, at de løser opgaver ved selv at udarbejde mindre programmer til datamaten, og at de anvender færdige programmer, der svarer til dem, som anvendes i arbejdslivet.

Konsekvens-afsnittet har som »Generelle synspunkter« den synsvinkel, at det er vore valg, der kan styre udviklingen:

En datamat er et generelt værktøj, som kan anvendes til mange endog meget forskellige opgaver, og hvor konsekvenserne ved brug af datamaten derfor mere er knyttet til den bestemte og konkrete anvendelse end til datamaten generelt. Edb er et redskab, der kan benyttes til mange forskellige, ofte modsat rettede formål. F.eks. kan en datamat bruges til at tilgængeliggøre information for mange i forbindelse med en decentralisering af beslutningsprocessen, men den samme datamat kan lige så godt benyttes til en centralisering ved at tjene til at koncentrere viden, magt og indflydelse på få hænder. Konsekvenserne af udviklingen på edb-området er således i høj grad et resultat af, hvorledes man beslutter at anvende denne teknik, hvor gode vi er til at anvende den og ikke af teknikken i sig selv.

Datamaternes »ufejlbarhed« får også et ord med på vejen:

En af konsekvenserne af den udstrakte brug af edb er en påvirkning af vor tænkning og vore handlingsmønstre. Vor virkelighedsopfattelse påvirkes, så det, der kommer på skærmen eller som edb-udskrifter, undertiden forveksles med virkeligheden. Der eksisterer her en risiko for, at man tillægger edb en slags ufejlbarlighed, og at man derfor ikke sætter spørgsmålstegn ved resultaterne. Herved får datamaterne en indirekte styrende funktion, som man skal være opmærksom på. Der kan ske det, at synspunkter og arbejdsformer bliver anset for mindre værdifulde, dersom de ikke grunder sig på elektronisk databehandling, men i stedet på f.eks. holdninger eller intuition.

Kapitel 3 omhandler undervisningens tilrettelæggelse, der eksemplificeres ved tre forskellige årsplaner og tre eksempler på konkrete undervisningsforløb. I de generelle bemærkninger om undervisningens tilrettelæggelse siges bl.a.:

Afsnit 3.2 indeholder eksempler på årsplaner, som hver for sig vil kunne danne grundlag for undervisningen i datalære i et étårigt forløb. Som det fremgår, er de valgte eksempler på årsplaner forskellige i deres opbygning. Dette illustrerer bl.a. den frihedsgrad, man som lærer har ved tilrettelæggelsen af undervisningen.

Den første årsplan er tilrettelagt således, at eleverne først arbejder med emner fra de to første hovedområder. På grundlag heraf belyses de to sidste hovedområder gennem arbejdet med et eller flere projekter. I den anden årsplan tages der udgangspunkt i de situationer, hvor eleven fremover vil møde informationsteknologien, mens den tredje er bygget op omkring karakteristiske anvendelser af edb.

For at give et indtryk af, hvorledes de fire hovedområder kan støtte og belyse hinanden, bringes der i afsnit 3.3 fra hver af årsplanerne et eksempel på en detailplanlægning af et mindre undervisningsforløb.

Undervisningsvejledningen peger altså ikke på en ganske bestemt måde, som undervisningen skal tilrettelægges på, men åbner for flere muligheder. De tre årsplaner med tilhørende eksempler på undervisningsforløb giver en god støtte og vejledning til lærere, der skal starte med en datalæreundervisning, men de indeholder samtidigt en inspiration til de mere erfarne lærere; på en måde signalerer forskellene mellem de tre eksempler nemlig, at andre angrebsvinkler også vil være mulige.

Det vil føre for vidt at medtage eksemplerne på årsplanerne i fuldt omfang, men et par klip fra bemærkningerne er måske på sin plads.

Strukturen i det første eksempel fremstår således:

Årsplanen er udarbejdet under hensyntagen til en bestemt organisation af undervisningen, hvor hovedvægten fra undervisningens start lægges på faglige emner fra hovedområderne kommunikation, information og data, problemløsning med brug af datamater. Gennem året forskydes vægtingen, så behandlingen af emnerne fra hovedområde 3 og 4 drages mere og mere ind. Afslutningsvis arbejdes med et større projekt, hvis tidsramme kan udgøre op mod halvdelen af den samlede undervisningstid, og hvor elementer fra alle fire hovedområder indgår i et integreret forløb.

Problemløsning og rutediagrammer får også et ord med på vejen:

Problemløsning indgår i alle undervisningens faser i form af opgaver, der belyser eller udspringer af behandlingen af de øvrige emner.

Det er ikke meningen, at eleverne skal bruge megen tid på formelle aktiviteter omkring f.eks. systemkonstruktion, rute- og tilstandsdiagrammer m.v. Sådanne emner medtages kun i det omfang, de skønnes bedst egnede til at skabe en øget forståelse.

Indholdet i det andet eksempel er opdelt i følgende hovedemner:

- A. Teknikken og hjemmelivet.
- B. Teknikken og kulturlivet.
- C. Teknikken og arbejdslivet.
- D. Teknikken og kommunikation.
- E. Den enkelte, teknikken og samfundet.

Hvert hovedemne er så opdelt i en række delemner og der sluttes med følgende bemærkning:

Udgangspunktet for arbejdet er ofte en begivenhed, der er behandlet i massemediernes – eller et forløb, en sag eller en begreb fra elevernes hverdag eller omgivelser. Hvor det er muligt, indlægges der konkrete øvelser og opgaver, som eleverne skal løse på datamaskinen. I nogle

tilfælde drejer det sig om at benytte færdige programmer, f.eks. afprøvning af modeller, eller programstumper. I andre tilfælde kan eleverne alene eller i samarbejde med en lærer udarbejde mindre programmer, der belyser eller underbygger det delemne, der arbejdes med i undervisningen. I begge tilfælde skal eleverne arbejde programmerne i forbindelse med den konkrete problemløsning.

Det tredje eksempel giver på listeform – som direkte inspirerer til, at den enkelte lærer fortsætter og forlænger listen – en række konkrete anvendelser af edb, hvilke større anvendelsesområder disse kan henføres til og peger endelig på nogle af de emner, der kan tages op i forbindelse med behandlingen af konsekvenserne.

Kapitlet slutter med nogle betragtninger om lokaler, holddannelser og undervisningsmidler. Der peges bl.a. på, at det vil være hensigtsmæssigt med et særligt lokale til datalære, så udstyret kan være tilgængeligt også til andre formål, og det fremhæves, at der foruden arbejdspladserne ved dataudstyret også skal være adgang til almindelige arbejdsborde og at der skal være sædvanligt udstyr, som i et klasselokale. Med hensyn til selve dataudstyret har undervisningsvejledningen følgende bemærkning:

Af formålsbestemmelsen fremgår det, at undervisningen skal give eleverne mulighed for oplevelse af og erfaring med problemløsning gennem brug af datamater.

Ved undervisningsmidler til faget datalære vil man forstå enhver form for udstyr og hjælpemidler, der bidrager til at opfylde formålet, herunder ikke mindst formålets stk. 2. Det omfatter således selve datamaten med indbygget eller tilsluttet disktestation samt printer og herudover ekstraudstyr som plotter, procesudstyr, »skildpadde« m.v. samt nødvendigt programmel.

Med henvisning til læseplanen kan det slås fast, at datamater er et uundværligt udstyr. Derimod stilles der ikke på grundlag af læseplanens forslag krav til omfanget af udstyret, til den kapacitet, det i de forskellige henseender skal være i besiddelse af eller til mængden af arbejdspladser. Dette afgøres ud fra den valgte pædagogik.

Sammenfattende kan man sige om undervisningsvejledningen, at der er nogle gennemgående principielle anskuelser, som beskrives flere steder. Dette gælder f.eks. programmering, hvorom det klart slås fast, at programmering skal med i datalære – også med en rimelig stor vægt. Men samtidig placeres programmering også klart som et middel: Det er ikke målet for datalære at uddanne edb-specialister.

Et gennemgående træk er også en åbenhed, der inspirerer og opfordrer til at prøve forskellige veje. Undervisningsvejledningen er ikke en spændetrøje, der skråsikkert anviser hvad der er rigtigt og hvad der er forkert, men mere en ramme, der trækkes op af de principielle anskuelser, og som indeholder mange ideer, synsvinkler, og forslag.

Udviklingen af datalære er ikke afsluttet med denne undervisningsvejledning – tværtimod . . .

red.

SFU - INFORMATION

BØGER MED PROGRAMMER

MATERIALER TIL GYMNASIETS 30-TIMERS KURSUS OG LÆRERNES 40-TIMERS KURSUS I EDB

COMAL I OVERBLIK

af Niels Bandholm & Jørgen E. G. Nielsen

COMAL I OVERBLIK er et introduktionskursus i EDB. Datamaskinens opbygning og funktion beskrives ved simple analogier. I et øvelsesprogram lærer brugeren selv at betjene datamaskinen og udforme enkle grafiske programmer. Til dette er udarbejdet en diskette med mere end 30 programmer, hvoraf nogle går videre end bogens tekst.

Øvelsesprogrammet, der er LOGO-inspireret, er udformet, så det apellerer til brugerens fantasi og kreativitet. Fra starten arbejdes med færdige programmer og gennem øvelserne læres stadig flere af COMAL-sprogets elementer, således at brugeren til sidst er i stand til at læse COMAL-programmer, foretage rettelser i sådanne og selv konstruere enkle programmer.

Øvelsesprogrammet bliver perspektiveret gennem bogens indledende og afsluttende del, således at COMAL I OVERBLIK kan give appetit på videre studier i informationsteknologien.

Maskintyper: Comet/Ditamat, Piccolo/RC 700, Piccoline, Butler, Commodore 8032 og Commodore 64. Forventes også leveret til Apple.

40 sider 16,5x23,5 kr. 35,25

Diskette kr. 395,00.

SULTEGRÆNSEN er et undervisningsspil, der behandler sammenhængen mellem befolkningsudvikling og fødevarerproduktion i de fattigste u-lande. Undervisningsforløbet spilles i dialog med datamaskinen. Spillet har den fordel, at modellen er så gennemsigtig, at det er muligt at følge med i dens beregninger og konsekvenser. Datamaskinen optræder ikke som en uigennemskuelig beslutningstager, men som et naturligt værktøj. Programmet er ligeledes udformet på en sådan måde, at det i praksis vil opfange alle fejlsituationer, hvad enten de skyldes fejl-

indtastninger eller misforståelser. Dette sammen med den udførlige brugervejledning - gør spillet let at gå til, selv for den EDB-ukyndige lærer eller elev.

EDB-version af SULTEGRÆNSEN

af Jørgen E. G. Nielsen

EDB-VERSION AF SULTEGRÆNSEN leveres i et praktisk, stift omslag med forside og indlagt orientering om programdisketten. Omslaget har samme format som bogen, hvilket er praktisk ved opbevaring på en reolhylde.

Programmet er udviklet i PASCAL og kan anvendes på maskintyperne COMET/Ditamat, Piccolo/RC 700, Piccoline og Butler.

Forventes også leveret til Apple og IBM PC.

Pris: Programdiskette + 8 stk. EDB-VERSION AF SULTEGRÆNSEN 570 kr.

DEMOGRAFIK af Karl-Erik Christensen

DEMOGRAFIK er meget fleksibelt og brugervenligt. Med brug af højopløsnings-grafik behandles demografiske emner - udvikling i indbyggerantal, alderspyramider og befolkningskurver. Egne indtastede data gemmes i en database.

Programmet er et pædagogisk og elevenligt arbejdsredskab i fagene geografi, historie, samfundsfag og samtidsorientering til behandling af emner inden for lokalhistorie, fysisk planlægning, regional udvikling og demografiske analyser.

Til disketten hører en udførlig brugervejledning med forslag til undervisningsforløb tillige med en kommenteret litteraturliste. Materialet kan benyttes i gymnasiet og HF såvel som i folkeskolens ældste klasser og på seminariet.

Maskintyper: RC Piccoline og Commodore 64.

20 sider 16,5 x 23,5 - kr. 23,00. Diskette: kr. 495,00.

Alle priser er excl. moms og forsendelse.

FLEX af Bjarne Tams & Finn Terpling

FLEX er et generelt værktøj til at opsøge og behandle informationer på tekstform. FLEX er generelt, fordi brugeren selv forfatter de informationer, der ønskes tilgængelig via systemet. Systemet er menudrevet og selvforklarende under brug. Ved hjælp af systemets forgrænsningsstruktur bliver man hurtigt ledt til de søgte informationer. Samtidig er systemet velegnet til at præsentere viden i en overskuelig form.

På programdisketten medfølger omfattende indlagte tekster til faget religion. De indlagte tekster definerer grundlæggende begreber inden for kristendom, islam, az'tekisk religion. Bruget af FLEX støttes af et udførligt bogmateriale på 40 sider med brugervejledning og tilhørende opgaver.

Maskintyper: RC Piccoline. Forventes også leveret til IBM PC.

Gymnasiet, HF, folkeskolens ældste klassetrin og seminariet.

40 sider, 16,5 x 23,5 - kr. 36,00.

Diskette: kr. 1.200,00 (til undervisningssektoren). Alle priser er excl. moms og forsendelse.

FORLAGET SFU

SIMULATIONS-FORMIDLET UNDERVISNING
Klokkerfaldet 88, 8210 Århus V. • Tlf. (06) 15 83 40

Kurset i Nakskov

Af Anne Mette Ejsing, Bramminge.

I dagene den 26.-28. april 1985 afholdtes Datalæreforeningens folkeskolefrakton sit årlige forårskursus i Nakskov. Hovedtemaet var: Datamaten i folkeskolen.

Ved ankomsten til hotel Harmonien blev vi installeret på gode værelser, de fleste med bad og toilet. Efter aftensmaden fordeltes vi i de valgte workshops, hvor der blev arbejdet de næste par timer, for nogle deltagere blev det til en del mere.

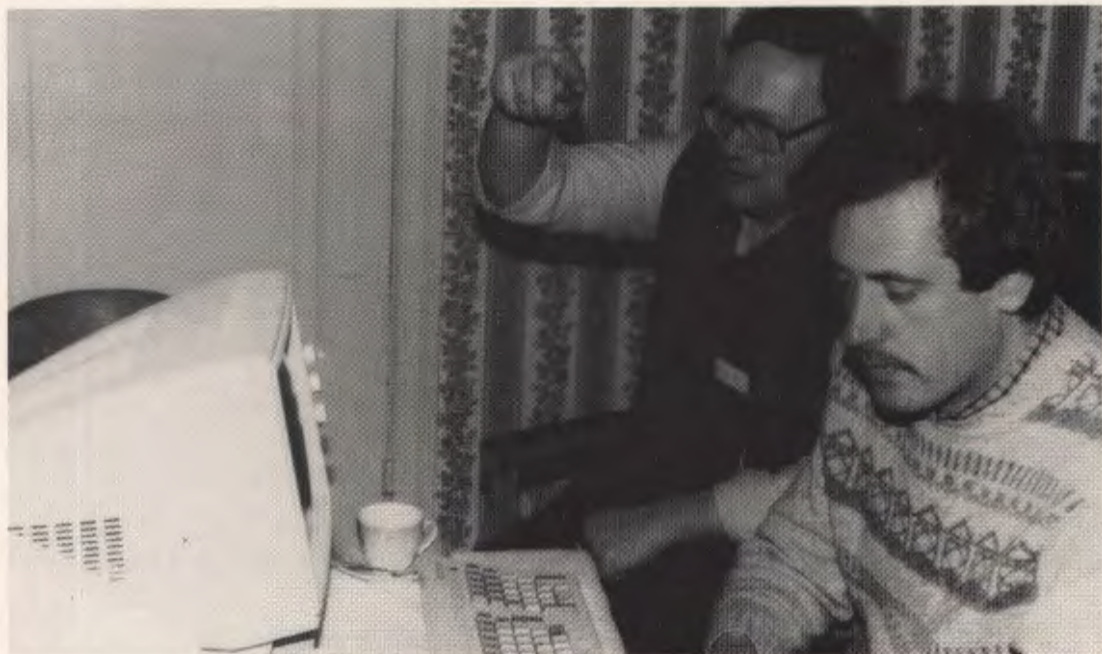
Lørdag morgen fortsættes arbejdet i andre workshops, og efter en god frokost kom Erling Schmidt med nyt fra Datalæreudvalget. De fleste af os havde nok håbet at få forslaget til undervisningsplanen at se, men da den endnu ikke var godkendt af udvalget, nøjedes Erling Schmidt med at fortælle om indholdet og læse passager op herfra, og han gjorde i det hele taget status for udvalgets arbejde. Han sluttede af med at vise og fortælle om, hvor faget datalære står i folkeskolen nu, gennem en tegneserie med Lucky Luck (Lucky Luck = datalærer), som hans vanskelige vej op til bjergets top, og hvorledes han kommer ned igen (flere muligheder). Mange ville sikkert gerne have haft fingre i denne serie, men hvad ville den være uden Erling Schmidts måde at sige tingene på. Der var stort bifald til E.S., for hans indlæg.

Senere på eftermiddagen og søndag formiddag var der igen workshops. Man kunne deltage i fire emner. Chr. Wang stod for en af disse med måling, styring og regulering, lyskryds, tegnebil m.m. Som noget nyt havde han en synthesizer med, som kunne styres pr. datamat.

Fritz G. Knudsen tog sig af mikrologoen og fortalte om sit forsøg hermed gennem 2 år. Erling Schmidt introducerede det kommende kommunevalg ved at vise, hvorledes han med et datalærehold havde arbejdet med denne opgave for knap 4 år siden i Aalborg. Karl Johan Jørgensen lagde op til en debat om, hvorledes man kunne anvende datamaten som hjælpemiddel og støtte i den daglige undervisning, og viste programmer fremstillet til Odense forsøgene. Bjarne Belhage fortalte datamaten i sprog- og formningsundervisning på sin meget fornøjelige og engagerende måde. Flemming Holt havde en workshop over datalære på mellemtrinnet, sikkert med sin tændstikdatamat Skjold. I de to sidstnævnte workshops har jeg ikke deltaget denne gang og er derfor ikke i stand til at kommentere disse yderligere.

Aftenerne gik som sædvanligt med at lære hinanden at kende og udveksle erfaringer, ideer og programmer. Der var begge aftener og det meste af natten livlig aktivitet omkring datamaterne, dog ville det have været rart, hvis der havde været flere mærker repræsenteret.

Det var et godt arrangement, værelserne var gode, maden var fin, alt klappede, takket være Jens Krog, der listede rundt og fandt ledninger og andre ting og sørgede for, at alt var i orden. Det var synd, at kun knap 30 deltog i et veltilrettelagt og velfungerende kursus, som Jens Krog fortjener en stor buket roser for.



»Den red den vej«.

Hurra for datalæren - men hvilken?

Af Jens Peder Pedersen, Brundlundsskolen, Abenrå.

Nu er datalære en realitet i folkeskolen. Som en skiftevis voksende og svindende betændt byld har kravet om indførelsen af datalære/edb i folkeskolen rumlet i adskillige år. I de senere år er bylden blevet stadig mere betændt i takt med såvel den tekniske udvikling, som kravene fra kredse udenfor folkeskolen og kradserier udført i lærerkredse. Og omsider er der gået hul på den. En blanding af gammel råddenskab, pus og frisk blod vælter ud af den. Folkeskolen fik sin datalære, men hvilken datalære?

Datalære er indført som et muligt valgfag i 8.-10. kl. i folkeskolen. Men kun for en overgangsperiode. Senere skal datalæren være obligatorisk under en eller anden eller flere former på nogle endnu ikke udpegede klassetrin. Det ligger altså temmelig fast, at datalære skal eksistere indenfor fremtidens folkeskole. Men ingen aner noget om, hvad der i den forbindelse menes med begrebet datalære?

Lad det være skrevet med det samme: Jeg har selv i de senere år med en blanding af spænding, glæde og afmagtsfølelse deltaget i ovennævnte kradseri. Jeg har været sikker på, at datalære var det helt rigtige i folkeskolen, og jeg har følt, at det hele var en stor misforståelse. Med fryd, irritation og afsky har jeg hørt/læst på profeters udgydelser i denne sammenhæng. Efterhånden udkrystalliseres det store spørgsmål: Hvad er det vigtige, som alt dette drejer sig om? Hvilken datalære?

Datalære kan sagtens gennemføres uden at eleverne ser en eneste datamat. Begrundelse: Datalære er læren om algoritmer / datalære er læren om magt / datalære er læren om samfundsudvikling. Eller: Datalære kan kun gennemføres ved hjælp af datamater. Begrundelse: Datalære er at anvende datamater / datalære er at programmere. Eller: Datalære hører slet ikke hjemme i folkeskolen. Begrundelse: Det er synd at sætte eleverne overfor datamater / datamater er et onde, der bør bekæmpes. Alt efter behag!

Hvilken datalære gennemføres mon i praksis i disse år? Se i læseplanerne, siger nogen. Dog gad jeg nok se den skole, der kan gennemføre datalære i overensstemmelse med den der vedtagne læseplan for datalære uden at ty til grove fortolkninger. Men lad nu det være. Her er jo blot tale om en kort overgangsløsning for de i øvrigt alligevel fortabte årgange. Lad os hellere se på fremtiden. Kun der kan idealerne blomstre - i visionen.

Hvad er det vigtige, der er indeholdt i begrebet data-

lære, og som folkeskolens elever i fremtiden skal modtage undervisning i? Er det programmering? Algoritmer? Politik? Samfundslære? Datamaten som hjælpemiddel? Etik? Kult?

Forhåbentlig er datalæren ikke det ene eller andet eller tredje, men derimod nogle væsentlige, grundlæggende aspekter af det basale i, at der i vor tilværelse af os er installeret et hav af tekniske mekaniske/elektroniske hjælpemidler, som vi er pint til at omgås, at beherske, at ligge under for, at betjene, at vedligeholde, at forbedre, at forstå.

Iblandt disse »apparater« findes computeren - ligesom kopimaskinen, radioen, bussen, blyanten, fyret, bruse- og vinduet i soveværelset.

Mange af disse ting har vi levet med i årevis. Ikke sådan, at vi bare har accepteret deres tilstedeværelse, næh, vi har anvendt dem, udviklet dem eller udskiftet dem med andre og bedre, mere fikse eller mindre generende.

Hvor har folkeskolen været henne i alt dette? Folkeskolen har som alle andre personer/institutioner brugt apparaturet, der hvor det fandtes »naturligt«. Derudover er der oprettet nye fag / nye undervisningsemner, der er gledet ind i skolehverdagen og nu findes at være traditionelle. Regning blev algebra, statistik, sandsynlighedsregning og geometri, sang blev musik, fysik blev bevægelseslære, elektromagnetisme, kernefysik og kemi, dansk fik sit udvidede tekstbegreb, o.s.v., o.s.v.

Der er sket en grundig revision af fagenes indhold og metoder, ligesom en række tekniske hjælpemidler er gledet ind i »naturlige« sammenhænge.

Op nu står vi overfor computeren, der kan synes at være en ganske ny og underlig tingest, så fremmed, at der naturligvis må oprettes et nyt fag, for at vore elever kan komme fremmedgørelsen til livs? - Blev der oprettet nye fag, da lokumsspandene afløstes af mekaniske toiletter, da hestevognene afløstes af biler, da budstikken afløstes af telefonen?

I forbindelse med de kulturelle ændringer, ovennævnte tekniske udviklinger medførte, blev der stillet nye krav til de bestående fag. Tidligere perifere/ligegyldige emneområder blev nu centrale. Det kan vel være, at man glemte noget eller ikke var forudseende nok. Det retter vi stadig på.

Er der i forbindelse med computeren så voldsomt nye ting i gærde, at en »naturlig« udvikling af de bestående fag, såsom dansk, orienteringsfagene incl. kristendoms-

kundskab, matematik og fysik/kemi, ikke kan magte det. Jeg tror det ikke. Og skulle jeg tage fejl, vil jeg foreslå, at vi ikke blot klistrer endnu et fag bag i rækken, men tager hele fagrækken op til grundig revision.

I første omgang vil jeg imidlertid i al beskedenhed fremsætte det forslag, at:

- 1) Computeren inddrages som hjælpemiddel i undervisningen, hvor det skønnes relevant.

- 2) De nye aspekter i elektricitetslæren overlades til fysik/kemi.
- 3) Algoritmerne overlades til matematik.
- 4) De etiske problemer sammen med øvrig orienteringsstof til orienteringsfagene.
- 5) Et valgfag, der ikke er forpligtet af alt det, der er overladt til punkterne 1-4, kan evt. eksistere på linie med sløjd, håndarbejde, m.m.

Hvorfor æder kaninen ikke pølsen?

– om brug af edb og logikprogrammering i forbindelse med sprog.

Af Keld E. Larsen, Amtsgymnasiet i Paderup.

Baggrunden for denne lille reportage er det 5-ugerskursus i edb, der netop (efterår 84/forår 85) er blevet afholdt for 30-40 gymnasielærere i Århus amt – forestået af Datalogisk Afdeling ved Matematisk Institut, Århus Universitet. En del af dette kursus var afsat til mindre projektarbejder, og en gruppe gik i lag med temaet »**Syntaks og Semantik**«. Foruden undertegnede deltog følgende i arbejdet: Lydik Garm (Randers Statskole), Else Weiss Hansen (Randers Amtsgymnasium), Aage Jørgensen (Langkær Gymnasium), Ellen J. Tandrup (Amtsgymnasiet i Hadsten), Reinhard Brose (Amtsgymnasiet i Skanderborg) og Kirsten Bech Hansen (Grenå Gymnasium). Inspirerende datalogisk assistance blev ydet af Erik Meineche Schmidt, Århus Universitet.

Teoretisk baggrund

Udgangspunktet for vores arbejde var en række overvejelser over grænsefladen mellem det naturlige sprog og programmeringssproget – mellem den åbne, flertydige og dynamiske sprogbrug mennesker imellem og det lukkede, entydige og rigoristiske sprogbrug i kommunikationen med maskinen. Vi kunne i den nye logikprogrammering iagttagelse bestrebelser på at nærme sig det naturlige sprog. Omvendt er det en kendt sag, at sprogkyndige lige siden oldtiden har ofret flid og energi på at formalisere og standardisere det menneskelige sprog ved hjælp af grammatikker og ordbøger.

Samarbejdet mellem sprogvidenskab og datalogi kom allerede i stand i årene efter sidste verdenskrig (indtil videre), og drejede sig i første omgang om at bryde militære koder. Allerede i 50'erne øjnene man dog muligheden for at lade computeren oversætte fra et sprog til et andet – in casu fra russisk til engelsk. Det er i denne sammenhæng den store lingvist og filosof Noam Chomsky udvikler fundamentet for den moderne såkaldte »ge-

nerative transformationsgrammatik«. Transformationsgrammatikken interesserer sig for vejen fra det »nøgne«, semantiske indhold til sætningens færdige sproglige form, når man producerer sprog – og den modsatte vej, når man forstår (analyserer) sprog. Denne vej beskrives som et sæt af transformationer/genskrivningsprocesser, der lader sig illustrere som trædiagrammer (jf. eksempler senere). En oversætterautomat kan ikke indskrænke sig til en ombytning af leksikalske størrelser. I princippet må den efterbe menneskets sprogproduktion og -reception. Den må altså arbejde sig fra den færdige sætning i sprog 1 tilbage til den underliggende struktur (som en slags afkodning) og derpå frembringe/genskabe den samme essens nu blot i det regelsæt, der kendetegner sprog 2 (en slags nykodning). Man står her over for uhyre komplicerede processer, der kun vanskeligt og delvist lader sig formalisere. Ganske vist er der allerede nu udviklet software, der er anvendelig i begrænsede sammenhænge som oversættere, råoversættere, støtte til oversættelser, men kvaliteten er efter sigende endnu så dårlig, at det egentlige gennembrud må siges at ligge forude – hvis det da overhovedet ligger inden for datalogiens rækkevidde!

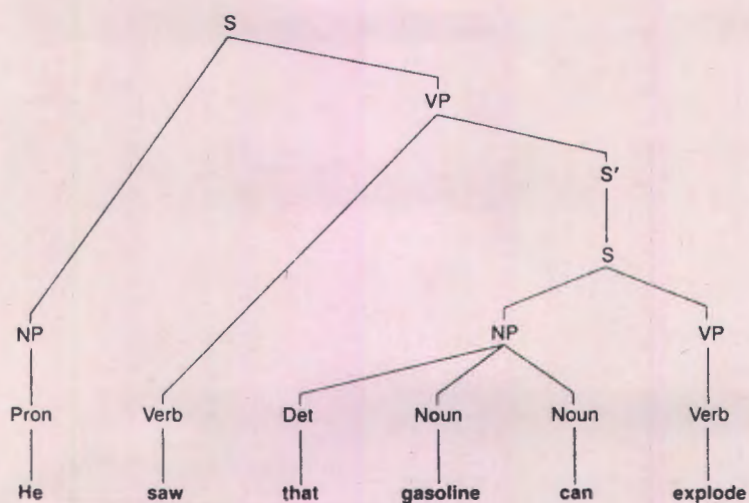
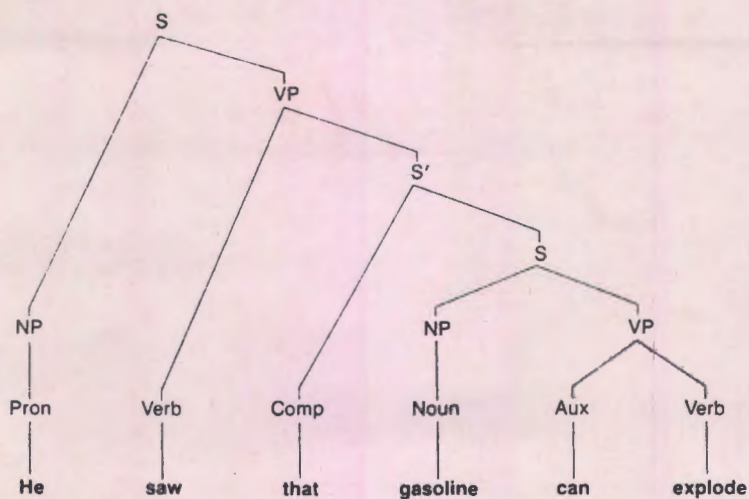
Om vanskelighederne ved at arbejde inden for dette område kan man læse i Terry Winograds artikel »Computer Software for working with Language« (Scientific American). T.W. fremhæver det naturlige sprogs **flertydighed** som det basale problem. Et enkelt ord kan være flertydigt:

Stay away from the bank!

flodbredden eller banken? Her bliver konteksten afgørende. Men problemet kan også ligge i sætningens grammatiske struktur:

He saw that gasoline can explode.

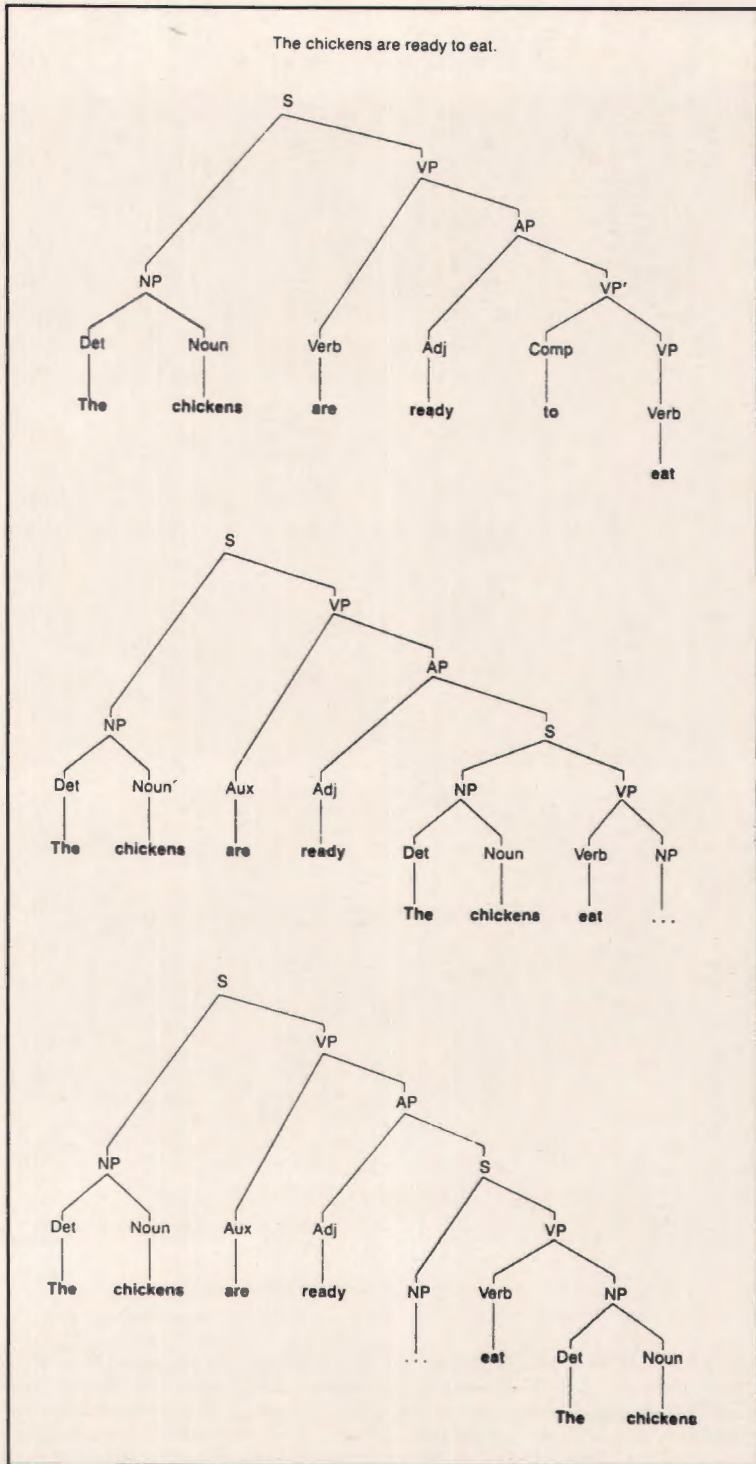
He saw that gasoline can explode.



STRUCTURAL AMBIGUITY arises when a sentence can be described by more than one grammatical structure. Here the conflicting possibilities for the sentence "He saw that gasoline can explode" are displayed in the form of grammatical "trees." In one of the trees the sentence has a subordinate clause whose subject is "gasoline" (*color*); the sentence refers to the recognition of a property of that substance. In the other tree "gasoline can" is part of a noun phrase (NP) meaning a container of gasoline; the sentence refers to the sight of a specific explosion.

Er »that« en bisætningsindleder (Comp) og »can« et hjælpeudsagnsord (Aux)?

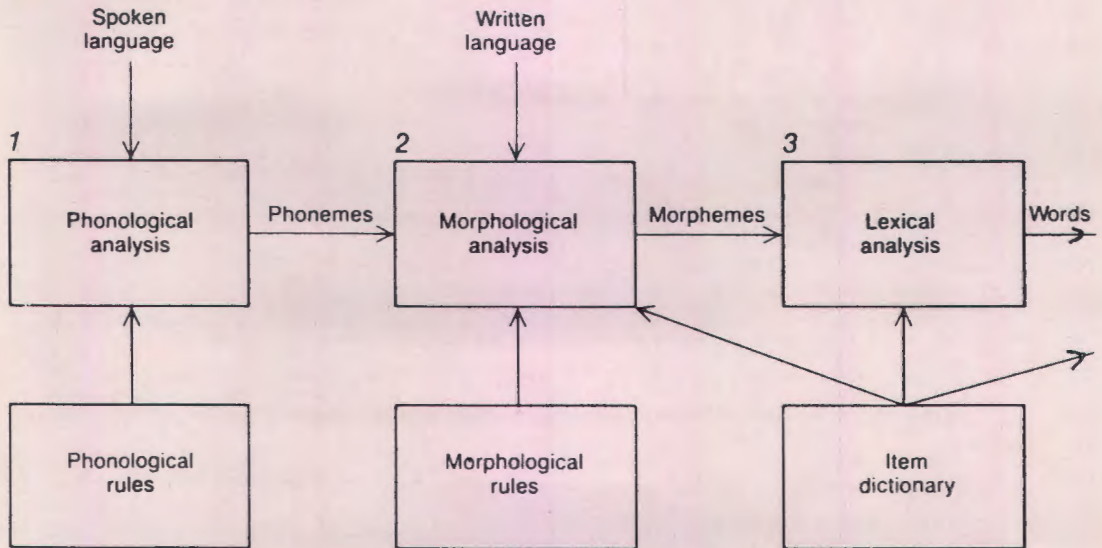
Eller er »that« et påpegende stedord (Det) og »can« et navneord?



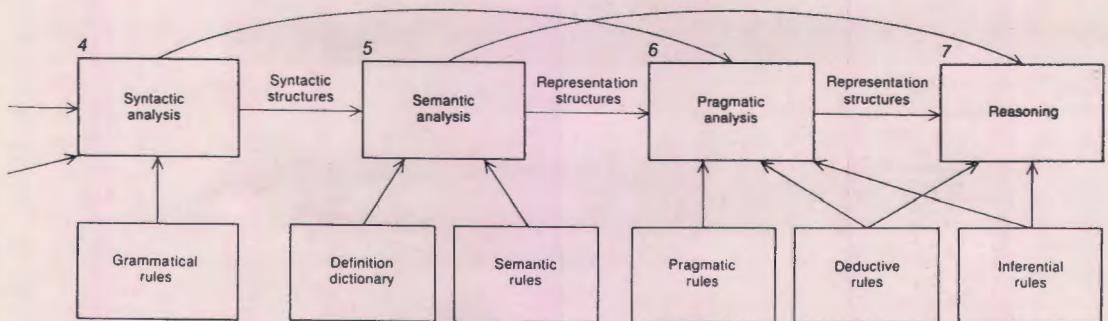
DEEP-STRUCTURAL AMBIGUITY arises when a sentence has a single apparent structure but nonetheless has more than one possible meaning. In this example the sentence is "The chickens are ready to eat." Its grammatical structure (*top*) leaves the role of the chickens ambiguous: in one interpretation they will eat; in the other they will be eaten. Deep-structure trees make the chickens' role explicit: they are the subject of the sentence (*middle*), in which case their food is undetermined, or they are the object (*bottom*), and their eaters are undetermined.

Dette blot som eksempel på de vanskeligheder, som datalogen løber ind i, når han vil dressere maskinen til at behandle naturligt sprog. Følgende model – ligeledes hen-

tet fra Terry Winograds artikel – giver et godt overblik over det apparat, der involveres for at få computeren til at »forstå« menneskets sprog:



COMPUTERIZED UNDERSTANDING OF LANGUAGE requires the computer to draw on several types of stored data (*white boxes*) and perform several levels of analysis (*colored boxes*). If the language is spoken, the first analysis is phonological (1): the computer analyzes sound waves. If the language is written, the first analysis is morphological (2): the computer decomposes each word into its root, or basic form, and inflections (for example *-ing*). Next is lex-



cal analysis (3), in which the computer assigns words to their lexical category (noun, for instance) and identifies "features" such as plurals. Then comes syntactic analysis, or parsing (4): the application of rules of grammar to yield the structure of the sentence. After that comes semantic analysis (5). Here the sentence is converted into a

form that makes it amenable to inference-drawing. The final stage is pragmatic (6): it makes explicit the context of the sentence, such as the relation between the time at which it is spoken and the time to which it refers. The computer is now in a position to draw inferences (7), perhaps in preparation for responding to the sentence.

Et indblik i moderne »computergrammatik« kan man få i Terry Winograds bog »Language as a Cognitive Pro-

cess. Volume 1: Syntax«. (Addison-Wesley Publishing Company).

Eksperiment i det små

I stedet for at læse mere om vanskelighederne besluttede vi nu i gruppen at nærme os dem fra den praktiske side - at lege os ind i dem.

Til det formål opstillede vi et lille sprogligt univers, som vi ville forsøge at gennemformalisere, så en maskine ville være i stand til at skelne mellem korrekte og ukorrekte danske sætninger - og dermed også i stand til at producere dansk sprog. Det var os allerede fra starten klart, at dette univers på den ene side måtte rumme nogle generelle problemer, men at det på den anden side måtte være dybt retarderet - ellers kunne vi ikke mestre det!

Resultatet blev (efterhånden) et univers på 20 ord, heraf 10 udsagnsord i nutid og 10 navneord i bestemt form: Med sans for det elementære og væsentlige i denne tilværelse defineres nu følgende univers:

elsker	kvinden
spiser	manden
æder	barnet
føder	hunden
kører	kaninen
lægger	bilen
ligger	kiksen
hjælper	pølsen
giver	spinaten
ruster	huset

Vi bestemte os nu for, at maskinen skulle lære at producere sætninger med op til 4 ord. Hermed var umiddelbart afstukket en kvantativ ramme på ca. 160.000 mulige kombinationer. Det turde være klart, at der imellem alle disse kombinationer kun var få acceptable danske sætninger - men hvordan skulle vi finde frem til dem? Vi var mange sprog- og dansklærere i gruppen (og de andre var da heller ikke direkte analfabeter!), så vi kunne hurtigt opstille en række restriktioner. Dog lå der stadigvæk et problem i at finde et velegnet programmeringssprog. Til vores formål blev vi anbefalet det nye logikprogrammeringssprog, Danlog (Kurt Fleckner og Jan Rubæk Pedersen), og under vejledning gik vi i gang med at konstruere programmet. Vores univers blev indlæst, og de nødvendige regler blev formuleret. Det var en klar fordel ved systemet, at disse regler til enhver tid kunne afprøves på universet, således at vi hele tiden var klar over, hvilke sætninger maskinen nu kunne producere.

Syntaktiske restriktioner

De første begrænsninger af universet var af rent syntaktisk art. I vores materiale består acceptable danske sætninger af 4 pladser, hvoraf de to sidste kan være tomme. Sætninger skal indeholde ét og kun ét udsagnsord, som står på 2. pladsen (vi har set bort fra spørgsmål). Endelig accepterer vi ikke, at et navneord forekommer flere gange i samme sætning.

n	u	(n)	(n)
---	---	-----	-----

Hermed er størstedelen af fejlmulighederne bortsorteret. Arbejder man nu videre med udsagnsordene, kan de tildeles ganske bestemte syntaktiske egenskaber:

a) nogle er intransitive (dvs. ÷ objekt)

n	u intr.		
---	------------	--	--

b) nogle er transitive (dvs. 1 objekt)

n	u tr.	n	
---	----------	---	--

c) nogle kan være både a) og b)

d) nogle er bitransitive (dvs. 2 objekter)

n	u bitr.	n	n
---	------------	---	---

Nu nærmer vi os noget, der ligner danske sætninger, men på skærmen dukker stadig betænkelige størrelser op:

bilen elsker
kiksen spiser kvinden
pølsen giver manden hunden

Vores kendskab til moderne digte og gamle eventyr sagde os, at sådanne sætninger sagtens kunne forekomme - på den anden side kunne vi dårligt se bort fra, at disse sætninger måtte siges at overskride en normalsprogsnorm. Her opstod det første principielle problem: **Digterisk sprogbrug/billedsprog**. Kunne vi f.eks. acceptere

manden ruster

Ja, som billede på en mand i forfald - men ikke i egentlig betydning. Vi valgte at arbejde videre med »**egentlige betydninger**« - vel vidende, at vi nu netop havde likvideret nogle af de vigtigste sider ved det naturlige sprog: Kreativiteten, farvningen, ironien . . . Det var den første omkostning!

Semantiske restriktioner

For yderligere at kunne beskære det vildt bolmstrende sprog måtte vi altså lægge endnu et sæt begrænsninger på - denne gang af semantisk karakter. Huset kører ikke - og manden giver ikke kiksen noget som helst! - Udsagnsordene stillede åbenbart ganske bestemte indholdsmæssige krav til omgivelserne, men hvilke? Kunne der overhovedet opstilles generelle semantiske retningslinjer? Denne fase af arbejdet var givetvis den vanskeligste. Dels blev problemerne omkring en sætnings korrekthed her radikaliseret (der var ikke uden videre enighed i gruppen), dels opstod der metodiske problemer (hvornår er en regel hensigtsmæssigt formuleret?)

Her blot et rids af et enkelt aspekt:

For at kunne formulere udsagnsordenes semantiske krav til de omgivende navneord, må disse forsynes med prædikater. Udsagnsordene kan f.eks. stille følgende krav:

elsker: grundledet (1. plads) = [levende]
ruster: grundledet = - [levende]
giver: grundledet = [levende]
 hensynsleddet (3. plads) = [levende]
 (i vores univers)

Men kan en kanin give noget?

Og kan et menneske give en kanin noget?

For at løse dette problem, må man som underkategori til [levende] indføre kategorien +/- [menneskelig]. Reglen for »giver« må da formuleres således:

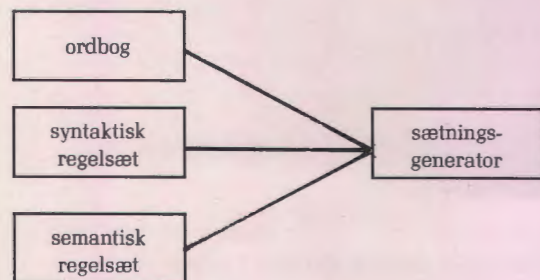
giver: grundled = [menneskelig]
 hensynsled = [levende]

Alligevel opstår der yderligere komplikationer, for hvis maskinen nu producerer en sætning med »giver« og et hensynsled, der nok er [levende], men - [menneskelig], så stilles der øjeblikkelig krav til genstandsleddet (4. plads), nemlig, at det ikke må være [menneskelig]. Manden kan ikke give hunden barnet! (Derimod kan kvinden godt give manden barnet). Her må der altså yderligere skrives ind med en sigtetsmæssig regel.

Jeg skal ikke her gennemgå alle de semantiske regler, men blot oplyse, at vores bud på en semantisk formalisering af vores 20-ordsunivers fylder godt og vel en side i programudskriften (men så vil den forhåbentlig også kunne dække et større univers).

Om Danlog-programmets opbygning og funktionsmåde

Det færdige program er principielt opbygget i følgende moduler:



Det skal fremhæves, at der under arbejdet med programmet skulle mange omarbejdnings til, inden vi fandt frem til en metode til at frembringe udelukkende korrekte danske sætninger. Men Danlog gjorde det på ethvert tidspunkt muligt at afprøve og forbedre reglerne. Kontrolmulighederne var fine, f.eks. gennem

- en listning/udskrift af de til enhver tid mulige sætninger
- en undersøgelse af et enkelt element i en given sætningsstruktur. (Eks.: Hvilke mulige grundled er der for udsagnsordet »æder« i 3-ordssætninger?)

- forespørgsel på en enkelt sætning. Spørger man Danlog, hvordan det givne resultat er opnået, vil man få en argumentation ved hjælp af de regler, man selv har lagt i programmet.

På denne måde var det let at justere programmet. F.eks. blev maskinen ved med at lade kaninen spise spinaten, indtil vi fik præciseret, at dyr (+ [levende], - [menneskelig]) »æder«. - Og hvorfor insisterede maskinen på at lade kaninen æde pølsen. Det gjorde den selvfølgelig, indtil vi fortalte den, at kaninen i modsætning til hunden er vegetar. - SÅ KUNNE KANINEN IKKE LÆNGERE ÆDE PØLSEN!

Som kuriosum kan det tilføjes, at vi ved projektets afslutning bed hovedet af al skam og konstruerede en oversætter (dansk/tyisk) til vores univers. Indgrebet kostede blot en ekstra spalte i ordbogen samt en række kasusregler, men det gav arbejdet en ny dimension.

Og hvad kan vi så bruge det til?

(spørger den trætte læser).

Det er vores tanke, at et forløb af denne karakter ville kunne anvendes i undervisningen og her åbne for et utraditionelt samarbejde mellem på den ene side dansk/fremmedsprog og på den anden side matematik/fysik. Det er forløbets pointe, at den pædagogiske gevinst ikke består i det færdige programresultat, men derimod i selve arbejdet med opbygningen af programmet og de tanker, som arbejdet iværksætter. Man kunne forestille sig et lignende forløb med en klasse med et lignende univers, men det materiale, som eleverne skal bearbejde, kunne ligeså godt handle om andre hjørner af sproget (f.eks. brug af stedord, tillægsord, biord, ordstilling, kasusstyring). Undervisningsudbytte kunne - afhængigt af forløbets udformning - bestå i:

- Øvelse i at formalisere på et utraditionelt område, nemlig sproget.
- Skærpelse af det sproglige opmærksomhed. Det kan her indskydes, at gruppens naturvidenskabelige lærere hævdede, at de havde lært en masse om grammatik, - og at de sproglige lærere gjorde gode erfaringer med at betragte stoffet på en i forhold til de traditionelle grammatikker uvant måde.
- Adækvat brug af maskinens ressourcer. Maskinen bruges som værktøj til at undersøge et område/en problemstilling. Hurtigt udregner maskinen kombinationsmuligheder og konsekvenser af regelsæt - noget der ville være svært og langsomt at foretage i hovedet eller på papiret.
- Gennem en sådan brug af maskinen befordres elevernes kreativitet. Via opfindsomhed, forsøg, revision og nytænkning konstruerer eleverne sig igennem emnet. Alt for ofte har vi som sprogslærere set undervisningsprogrammer, som egentlig blot kan betegnes som øvelsesark på skærm, og som hensætter eleven til en sekundær, reagerende, svarende rolle i forhold til maskinen. Gennem forløb som det, vi har været igennem, kan man - i det mindste på visse områder - etablere en grammatikundervisning, som er **induktiv**, og ikke

deduktiv som den meste grammatikundervisning i skolen i dag.

- 5) Indblik i forskellen på naturligt og kunstigt sprog. Eleverne vil kunne øge deres opmærksomhed omkring det menneskelige sprogs særlige kvaliteter – og samtidig lære noget om formaliserings effektivitet på den ene side og omkostninger (reduktionisme) og grænser på den anden side.
- 6) Indblik i et nyt og spændende område for den nye tek-

nologi: Oversætterautomaten. Der ofres rundt omkring i verden i disse år vældige summer på dette felt – også i EF-regi. Et anskueligt eksempel på, hvor langt man er kommet – eller hvor langt, man har igen, leverer det vesttyske ugemagasin »Stern«. Det drejer sig om en oversættelse af et stykke fra G. Orwells »Animal Farm« til tysk – henholdsvis i en computerudgave (»Systran«) og en litterær udgave:

Trotz Fehler – der Computer ist lesbar

Absatz aus Orwells

»Animal Farm« im Original

»Now, comrades, what is the nature of this life of ours? Let us face it: our lives are miserable, laborious, and short. We are born, we are given just so much food as will keep the breath in our bodies, and those of us who are capable of it are forced to work to the last atom of our strength; and the very instant that our usefulness has come to an end we are slaughtered with hideous cruelty. No animal in England knows the meaning of happiness or leisure after he is a year old. No animal in England is free. The life of an animal is misery and slavery: that is the plain truth.

Computer-Übersetzung

Nun, Kameraden, was ist der Sinn unseres Lebens? Lasst uns nachdenken es: unser Leben ist elend, mühsam und kurz. Wir sind geboren, wir bekommen gerade so viel Nahrung wie den Atem in unseren Koerpern zu halten, und jene von uns, die faehig es sind, werden gezwungen, das Letzte unserer Kraft zu geben; und genau dann, wenn unsere Nuetzlichkeit zu einem Ende gekommen ist, werden wir mit schrecklicher Grausamkeit geschlachtet. Kein Tier in England kennt die Bedeutung von Glueck oder Freizeit, nachdem er ein Jahr ist, alt. Kein Tier in England ist frei. Das Leben eines Tiers ist Elend und Sklaverei: dass ist die einfache Wahrheit.

Literarische Übersetzung

Nun, Genossen, wie ist die Natur dieses unseres Lebens? Seien wir ehrlich: unser Leben ist elend, mühevoll und kurz. Wir werden geboren, wir bekommen gerade soviel Futter, daß uns die Puste nicht ausgeht, und wer von uns dazu geeignet ist, wird gezwungen, bis zum letzten Deut seiner Kraft zu schuften; und just in dem Augenblick, wo es mit unserer Nützlichkei aus ist, werden wir mit scheuëlicher Grausamkeit hingeschlachtet. Wenn es erst einmal ein Jahr alt geworden ist, hat kein Tier in England mehr eine Vorstellung von Muëe und Glück. Kein Tier in England ist frei. Das Leben eines Tieres ist Jammer und Sklaverei: das ist die nackte Wahrheit.

PROJEKTER

i

DATALÆRE

med COMMODORE 64

Til hver af emnerne
REJSEBUREAU
FOLKeregISTER
PLADEKLUB

er udviklet programmer og opgaver.
Programmerne er skrevet i COMAL-80

PRIS pr emne: 350 Kr.

NÆRMERE OPLYSNINGER FÅS

hos

KG-DATA-Lunderskov

Knud Grosen

Odinsvej 10

6640 Lunderskov

Tlf. (05-585551)

BRUG SOMMERFERIEN TIL AT LÆRE LOGIK-PROGRAMMERING!



DANLOG

– et DANSK LOGIK-PROGRAMMERINGS-SPROG
af Kurt Fleckner
& Jan Rubæk Pedersen

DANLOG er det første dansk udviklede logik-programmerings-sprog til undervisningsområdet.

DANLOG er et dialog-orienteret femte-generations-sprog, der frigør brugeren for den detaljerede del af programmeringsarbejdet. DANLOG-systemet udmærker sig ved at være

et brugervenligt programmeringssprog med rette- og redigeringsfaciliteter på dansk. Systemet er menurevet ved hjælp af Piccolinens funktionstaster. Ved fejlmeldinger gives oplysende hjælpetekster. En »hvordan-knap« forklarer, hvordan DANLOG »tænker«. Systemet støttes af en lærebog med mange undervisningseksempler fra en bred fag række (eks. franske verber, DNA-kodning, symbolsk differentiation, reduktion af matematiske udtryk, stamtræ, sortering samt forskellige problemløsningsopgaver og - strategier). Disse eksempler findes også på disketten.

Maskintyper: RC Piccoline. Forventes også leveret til IBM PC.

Gymnasiet, HF, folkeskolens ældste klassetrin og seminariet.

100 sider, 16,5 x 23,5 - kr. 89,00.

Diskette: kr. 2.450,00 (til undervisningssektoren). Alle priser er excl. moms og forsendelse.

BRUGERSEMINAR I LOGIKPROGRAMMERING MED KURSUS I DANLOG

Logikprogrammering er grundlaget for fremtidens vidensbaserede systemer, som også tænkes anvendt i japanernes nye såkaldte »femte-generations« computere. Derfor er der væsentlige argumenter for at begynde at beskæftige sig med logikprogrammering.

FORLAGET S.F.U. og **HOBRO UDDANNELSESCENTER** arrangerer fredag 27. og lørdag 28. september et brugerseminar med indlæg af bl.a. lektor H. Siggaard Jensen, AUC som oplæg til kursus i DANLOG - det dansk udviklede logik-programmerings-sprog. DANLOG er et gennemprøvet system, der gennem 1/2 år har kørt på RC-Piccoline. DANLOG har været benyttet bl.a. på Århus amtskommunes datakursus for gymnasielærere i 1985.

Kursusprisen kr. 850,00 dækker kursusmaterialer, morgenmad, frokost, aftensmad, kaffe, lokaleleje, lærerkræfter og overnatning på Hotel Alpina, Hobro. Forlaget S.F.U. og Regnecentralen stiller lærere og maskiner til rådighed.

Undervisningen foregår på **HOBRO UDDANNELSESCENTER** fredag kl. 12.00-19.00 med adgang til maskinerne til kl. 23.00 og lørdag kl. 08.00-13.00.

VED KØB AF ET DANLOG-system + ET KLASSESÆT AF DET TILHØRENDE BOGMATERIALE ER DELTAGELSE I BRUGERSEMINARET GRATIS FOR ÉN PERSON!

Begrænset deltagerantal 16 pers.

FORLAGET SFU

SIMULATIONS-FORMIDLET UNDERVISNING

Klokkerfaldet 88, 8210 Århus V. • Tlf. (06) 15 83 40

Navn: _____

Skole: _____

Adresse: _____

Postnr.: _____ By: _____

Tlf.nr.: _____

Folkeskolefraktionens Nakskov-kursus 1985

Af Chr. Hansen, Vestenskov.

I weekenden den 26.-28. april var godt 40 medlemmer af Datalæreforeningens folkeskolefraktion samlet på hotel »Harmonien« i Nakskov.

Jeg var med på det første Nakskov-kursus i marts 1984 og var meget begejstret. Min begejstring for dette års kursus er ikke mindre.

Man har fundet en god form med et foredrag eller to og så ellers workshops, hvor kursusdeltagerne får mulighed for at afprøve forskellige projekter og ideer på de opstillede dataanlæg.

Men mellem disse faste og forprogrammerede kursuspunkter er der jo nogle pauser – bl.a. de første natteimer. I disse pauser, der er uhyre spændende, foregår der en masse. De enkelte deltagere udveksler tanker, ideer og erfaringer. Man diskuterer og argumenterer, men det mest spændende foregår omkring de mangeartede dataanlæg, som flere af deltagerne har medbragt – utroligt hvad der kan præsteres på disse maskiner, når de rigtige folk sidder ved dem.

Kursets eneste foredrag var overladt til Erling Schmidt først på eftermiddagen om lørdagen. Han gennemgik læseplanerne ved små kritiske og vittige »kig« i for- og fremtid.

Han ka' bare det der!

Erling er begejstret, inspirerende og meget engageret. Han fortjener megen ros for sit arbejde gennem de senere år.

Kurset omfattede 5 workshops, hvoraf man kunne vælge 4. Jeg fulgte ikke Fritz G. Knudsens Micro-Logo, da jeg var med på det første Nakskov-kursus og selv benytter Logo i det daglige arbejde i 4. klasse. Jeg kan forøvrigt varmt anbefale det. Sammen med børnene oplever man geometri på en anden måde.

Hos Karl Johan Jørgensen var emnet noget om datamaten i undervisningen. Her hørte vi om erfaringerne fra Odense suppleret med deltagerens erfaringer. En god, stilfærdig og saglig debat, som måske nok tog lidt for lang tid. Vi nåede ikke så meget af de programmer, som



»Og så gør man bare sådan«.



»Uden mad og drikke . . .«.

Karl Johan havde med - bl.a. kunne jeg godt have brugt lidt mere tid ved den datamaskine, som havde en »mus« tilsluttet. Det var sjovt, og jeg tror, at mange af os må se at få fat i den.

Flemming Holt gennemgik forsøget med obligatorisk datalære i 5. og 6. klasse på Gug og Gistrup skoler i Aalborg. Forståeligt nok var der nok at fortælle om og masser af spørgsmål, så også her blev der lige lidt tid nok ved maskinerne, hvor vi havde megen morskab med et program, der simulerede et supermarked. Der er udarbejdet rapporter over disse forsøg. Se venligst i sidste nummer af »Datalære«, side 30.

Hos Erling Schmidt var emnet kommunevalget. Dette program er udarbejdet ved skolevæsenet i Aalborg. I smågrupper har eleverne udarbejdet delprojekter, som så til sidst er blevet samlet til en helhed. Programmet blev afprøvet ved sidste kommunevalg, hvor skoleeleverne fik lov til at være med ved selve optællingen, hvor man så var først med udregningen af mandaterne. En oplagt idé nu op til efterårets valg.

Den 5. workshop blev ledet af Christian Wang. Han skal opleves - sprudlende fuld af humør og et fantastisk engagement. Hos ham styrer datamatens processer uden

for maskinen. Når disse Odense-projekter bliver almindelige i alle landets folkeskoler, så vil vi være nået et godt skridt fremad. Efter en hurtig, men grundig teoretisk gennemgang af de pricipper, der ligger til grund for de enkelte processtyringer, kunne vi så gå over til det praktiske arbejde, der er meget spændende og tidskrævende, således at de afsatte 2½ time føltes alt for korte. Vi arbejdede med tegnebil og lysregulering. Endvidere havde Christian en opstilling med synthesizer. Ganske vist kunne vi kun lave trommer, men det gik også fint. Vi nåede til nogle inciterende sambarytmer, og vi manglede egentlig kun nogle mørke brasilianske skønheder for at stemningen havde været fuldkommen.

Og så var klokken pludselig 12 - søndag. Det hele var forbi - ligesom man nu havde lært en hel masse at kende. Men sådan er det jo hver gang, man er på kursus. Fik man nu sagt pænt farvel og tak til alle?

Nåh! Jeg har jo her chancen til at få sagt tak for en god weekend. Tak til mine medkursister, til instruktørerne og sidst, men ikke mindst den stilfærdige og smilende »skygge«, der gled ind og ud af lokalerne og sørgede for, at alt forløb, som det skulle - tak Jens fra os allesammen.

Datalære i Grønland

Af Jytte Schmidt, Godthåb.

Er der datalære i det grønlandske skolevæsen?

Hvis der er maskiner, hvordan opfører de sig da, når luftfugtigheden er 35-40, når der er strømsvigt og aldrig svigtende statisk elektricitet?

Når den ene halvdel af eleverne kommunikerer på dansk, den anden på grønlandsk og maskinerne på dansk eller højniveausprog? Når skolevæsenet i tradition, princip og indhold bygger på det danske og Lærershøjskolen ikke er en by i Grønland?

Visse problemer egner sig ikke til at blive løst. Der er andre og mere påtrængende.

Jeg skal forsøge at give et rids af situationen som den ser ud for en dansk folkeskolelærer, der har været her i 9 måneder.

Også her er der 9 års skolepligt, obligatoriske fag, valgfag og tilbudsfag. Fagkredsen kan sammenlignes, men når obligatoriske emner som fiskeri, fangerlære, jagtvåbenlære, motorlære (bådmotorer) og fredning inddrages, må andet nødvendigvis ryge ud. 10.-13. klasse er den frivillige del af skolegangen. 11. og 12. klasse svarer til 9. og 10. klasse i Danmark. 1. klasse er her børnehavklassen.

Datalære indføres som valgfag i 86/87 for 10.-13. klasse, forberedelserne hertil er lige så forskellige som antallet af kommuner. Jeg har indskærket min research til Godthåb og vores nordlige nabo, Sukkertoppen. Hvad der undervejs er blevet sagt også om andre kommuner, bekræfter anelsen om, at Godthåb er den mest erfarne kommune. Det var da også her, det første kursus for lærere blev holdt. Det skete i 83/84 som kommunalt årskursus på 3 ugentlige timer med reduktion efter reglerne. De 105 timer gennemførtes af 14 lærere. 2 af dem underviste faget året efter.

Det er rimeligt at omtale kursets formål og indhold, da det i år (84/85) holdes i Egedesminde og det kommende år atter er i Godthåb. Underviseren var i 83/84 en matematiklærer fra Godthåb Seminarium og i 84/85 en lærer fra Egedesminde HF.

Kursusformål i 83/84: At give indblik i elementær brug specielt med henblik på undervisningen således, at forskellige fag får mulighed for at benytte maskinen som redskab og pædagogisk hjælpemiddel i undervisningen.

Indhold i 83/84: Datamaskinens opbygning, logiske strukturer og lagermedier. Problemformulering og opgavebeskrivelse. Programmer og afprøvning af programmer. Dokumentation.

Kursusformål i 85/86: At give indføring i den elementære brug af datamater med særligt henblik på folkeskolens kommende undervisning.

Indhold i 85/86: Datamaters logiske strukturer. Opgavebeskrivelse, programmering og afprøvning af programmer. Dokumentation.

Metode i 85/86: Specielt lægge vægt på at anskue datamaten som et værktøj, der via programmering kan benyttes til at løse mange forskelligartede opgaver.

Kurset i Godthåb gennemførtes på 6 Piccoloer, samlet på en af de 3 Godthåbskoler, men udgiftsfordelt på alle 3 (se fodnote 1). Valghold dannet af de 2 ældste klassetrin undervises i dette datalokale i år. Lokalet udnyttes fra kl. 8 til 17.30, dvs. 10 timer af 12 hold excl. et hold fra HF. De 12 hold undervises af 7 forskellige lærere. Nogle af holdene er rent grønlandske, andre er sammensat af elever fra danske og grønlandske klasser.

En egentlig læseplan har ikke foreligget, indhold og form har foreløbigt været baseret på det individuelle initiativ. Underviserens forudsætninger har været yderst forskellige. Den underviser, som var »supervisor« for fag og lokale, havde en svag, 10 år gammel baggrund, da han startede sine valghold dette år. Han kunne ikke søge 83/84-kurset, da han endnu ikke havde 2 års ansættelse her.

Som i det små, ligeså i det store: Kommunerne har fået frie hænder til indkøb, oprettelse, indholds- og omfangsbestemmelser af faget og til hovedjagt. Det har betydet, at nogle kommuner har afventet, de har brugt året til indhentning af oplysninger og til overvejelser, mens andre er gået i gang med risiko for at gøre erfaringer. Desværre er intet nedfældet, hvad også direktoratet må beklage. Der ligger dog et spinkelt datagrundlag i de pligtige pensaprotokoller, som indeholder oplysninger om materialer, fagindhold og lærerens vurdering af materialernes egnethed vedrørende ethvert fag. Pensaprotokollerne afleveres ved skoleårets slutning, så der vil ikke foreligge noget samlet overblik før i begyndelsen af det kommende skoleår. Eneste offentliggjorte data er altså kursusopslag og den her netop rundsendte læseplan for Københavns kommune med direktoratets supplerende kommentarer.

En af Godthåblærerne, som deltog i 83/84-kurset og i år har undervist, har som andre, jeg har henvendt mig til, venligst stillet sig til rådighed for interview. Erfaringerne gælder 2 hold på hver 12 elever fra 11. og 12. klasse. Det ene hold rent grønlandsk, det andet blandet. Årsplanen havde som sine 3 hovedpunkter at lære apparaturet at kende, programmering og virksomheders dataudstyr, anvendelse og brugererfaringer. Begge hold fik at vide, at dette var et begyndertilbud, undervisningen meddeltes på holdbasis uanset at nogle elever på det blandede hold var fortrolige med hjemmecomputere. For hver ugentlig dobbelttime havde læreren forberedt en bundet opgave, som behandlede det netop gennemgåede nye emne. Eleverne blev hørt i programmets opbygning og virkning. Efter den bundne opgave kunne de gå videre med deres frie opgaver. Et af virksomhedsbesøgene havde Handelsskolen som mål, her iagttog man bl.a. undervisning i edb.

Efter lærerens mening kom programmering til at optage for stor en del af den samlede undervisningstid. Skønt det var de ældste klassetrin, var det det faglige i bl.a. matematik, der forsinkede. Det grundlag, valget af data-lære blev truffet på, var ikke for alles vedkommende det allerbedste. Elever, der havde svært ved engelsk og tysk, tabere i systemet, så det nye fag som deres overlevelse og valgte det ud fra en forventning om, at det var den mindst krævende opfyldelse af minimumtallet for ugentlige timer. I løbet af året har 8 grønlandske elever meldt fra, mens alle danske er forblevet. En vigtig erfaring, som rækker langt ud over faget er, at blandede hold fungerer dårligt. På det grønlandske hold følte man sig tryk og stillede spørgsmål, det blandede hold var for ulige på for mange måder. Årsplanens hensyn til maskinstøttet undervisning generedes af det personlige synspunkt, at en sådan bør sortere under de enkelte faglærere. Disse havde jo også deltaget på lige fod i 83/84-kurset.

Andre Godthåblærere har gennemført undervisning på grundlag af en lærebog. En lærer, som har 2 danske hold, finder, at materialeuddet er kvantitativt for ringe. Eleverne har følgelig fortrinsvis programmeret. I begyndelsen anvendtes Datalære med Mikrodatamat fra System, også den tilhørende testdiskette blev brugt. Informatik I med de differentierede opgaver og løsnin-ger blev også brugt, men de tilhørende disketter var ikke anskaffet. Efter jul legede eleverne med Logo og enkelte spil. Frafaldet var kraftigt. Denne lærer havde ønsket sig nogle færdige programmer til sin undervisning. Bestanden af Piccoloer er netop udbygget med nr. 7 og 8. Det er endnu usikkert, hvor de skal anbringes endeligt. De 2 valfartende skoler forudser, at man fortsat må supplere med autodidakter i faget. Det er derfor vanskeligt at afslå en enkelt maskine pr. skole.

Om fagets fremtid nøjes man ikke med at spå. Direktora-tet har til intern brug udarbejdet et handlingsprogram,

som bl.a. omfatter en konference til efteråret. Foreløbig står 5 punkter på programmet:

1. Forslag til formål og læseplan.
2. Forslag til maskintyper og antal.
3. Forslag til programmeringssprog og stilling til central programsamling.
4. Forslag til lærerkurser.
5. Muligheder for elektronisk formidlet undervisning.

Direktoratet anbefaler foreløbigt den københavnske læ-seplan for sin rummelighed og knappe længde. Endvi-dere ønsker man eksempler på datastøttet undervisning indarbejdet i de enkelte fags vejledninger.

Konferencen vil samle repræsentanter for de grønland-ske kommunale skolelæesener og indkalde specialister udefra f.eks. fra Dansk Center for Informatik og Pædago-gik.

Kommunikation og distribution er naturligt besværligt i Grønland. De små isolerede enheder begrænser. Kun byer som Godthåb, Egedesminde, Julianehåb, der har HF, kan gøre sig håb om lærerkræfter til kursusledelse. Kommunerne må senest i 85/86 købe maskiner. Juliane-håb har købt Vic 20, Narssaq Commodore 64, Godthåb Piccolo, (seminariet en Piccoline) og Sukkertoppen havde et beløb, men bestemte sig for indkøb af ski i ste-det. I Sukkertoppen er faget planlagt undervist i 85/86. Ingen af de nuværende lærere kan forestå undervisnin-gen. 2 lærere går på aftenskole og vil til vinter have af-sluttet 1. del af merconomkurset. De undervises på lære-rens private udstyr af mærket Sinclair. Valget vil blive kaldt regnskabsføring via datalære. Enkelte lærere har en hjemmecomputer. Uummanaq har ikke købt, men forhørte sig telefonisk hos en Godthåblærer, der fandt at RC's filial i Jakobshavn havde hjulpet ham så godt, at han ville anbefale det mærke.

Jeg vil gerne skrive næste afsnit i føljetonen, når konfe-rencen til efteråret har afklaret forskelligt.

Jeg skylder adskillige stor tak for den tid og venlighed, de har ofret på mig i anledning af denne artikel. Unøjagtig-heder tilskrives min ringe opfattelsesevne. Som før nævnt er de fleste oplysninger overleveret mundtligt.

Min tak gælder især:

Ib Follin, Direktoratet for Kultur- og Undervisning, Grøn-land, Daniel Skifte, Kulturel Forvaltning, Godthåb, Leif Aidt, Atuarfik Samuel Kleinschmidt, Godthåb, Margit Motzfeldt, Qorsussuaq, Godthåb, Knud Lindelof, Uka-liussaq-Skolen, Godthåb, Peter Morville, Atuarfik Kilaa-seeraq, Sukkertoppen.

Jytte Schmidt, Piccolineer, serienr. 95.

1) En redegørelse for vedligeholdelsesudgifter og service på hysteriske Pic-coloeer m.fl. hører til i en artikel for sig.

((PROLOG)(DATA))

((PROLOG)(DATA))

PROLOG opstod i begyndelsen af 70'erne, og har siden fastslået sin position som fremtidens programmeringssprog. PROLOG fås i dag til alle typer datamater, fra hjemmecomputere over mini- og micro-datamater til mainframe-anlæg.

μ

micro-PROLOG

micro-PROLOG har en syntaks, der gør det muligt at kommunikere i et næsten naturligt sprog. Samtidig er det dog muligt at bruge Edingburgh-syntaksen.

apes

apes

apes er en udvidelse af micro-PROLOG, der gør systemet interaktivt og fuldt argumenterende. Det er velegnet til konstruktion af intelligente databaser og eksperter-systemer.

Σ

sigma-PROLOG

sigma-PROLOG er specielt udviklet til brug under 68000 UNIX og DEC-VAX. sigma-PROLOG er en yderst avanceret Prolog, der har faciliteter, som ikke findes i nogen anden Prolog til UNIX-maskiner.

プロロ- 京都

prolog-KABA

Prolog-KABA er en usædvanligt hurtig PROLOG fra Japan, der er blandt de føende lande indenfor udviklingen af og forskningen i logikprogrammering. Prolog KABA benytter Edingburgh syntaks.

micro-PROLOG fra Logic Programming Associates kan leveres til maskiner med operativsystemerne CP/M-80/86 og PC/MS-DOS.

micro-PROLOG kan fås i specialversioner til McIntosh, Spectrum, Commodore, Amstrad m.fl.

Priser:
 CP/M-80 (min 48 KB) kr. 1300
 CP/M-86 og PC/MS-DOS (min 128 KB) kr. 3600
 McIntosh kr. 5200
 Professional MS/DOS (min 256 KB) kr. 7300
 Hjemmecomputere spørg

apes fra Logic Based Systems kan leveres til operativsystemerne CP/M-86 og PC/MS-DOS.

Priser:
 apes (min 128 KB) kr. 3600
 apes Professional MS-DOS (min 256 KB) kr. 7300
 micro-PROLOG + apes (min 128 KB) kr. 8500
 micro-PROLOG + apes Professional kr. 12000

sigma-PROLOG fra Logic Programming Associates kan leveres til 68000 UNIX og DEC-VAX.

Priser:
 68000 UNIX (min 0.5 MB) kr. 12000
 DEC-VAX (min 0.5 MB) kr. 25900
 DEC-VAX: service og opdatering, årligt kr. 12900

Prolog-KABA fra Kyoto Artificial Brain Associates kan leveres til maskiner med operativsystemerne CP/M-86 og PC/MS-DOS.

Pris:
 (min 256 KB) kr. 7200

Sammenligning af forskellige PROLOG-sprog til microdatamater, foretaget af Science University of Tokyo ved Prof. P. Mizoguchi. (The Logic Programming Conference, Tokyo marts 1984):

sprog	operativsystem	100 kørere	
		list reverse	quicksort
		30 elementer	50 elementer
Prolog-KABA	PC-DOS, 5.0 MHz	0.722 sek.	0.920 sek.
micro-PROLOG	CP/M-86, 4.9 MHz	8.33 sek.	7.32 sek.
PROLOG-1	CP/M-86, 4.9 MHz	17.3 sek.	11.1 sek.

LITTERATUR.

micro-Prolog Fakts og regler	J. Briggs	Myntevej 3 9380 Vestbjerg (08) 296124
micro-PROLOG: Programming in Logic Programming in micro-PROLOG	K.L.Clark & P.G.McCabe	
Beginning micro-PROLOG 2.rev.ed.	H.de Saram	
Problem Solving with micro-PROLOG	R.Ennals	
Implementations of PROLOG	P.Hepburn	
An Advanced Course in PROLOG Programming	J.A.Campell	
An Introduction to LISP	L.A.Spacek	Rørdalsvej 100 9200 Aalborg Ø
Communicating with Databases in Natural Language	A.Narayanan	(08) 135035
Automatic Natural Language Parsing	M.Wallace	Kong Christians Alle 26 9000 Aalborg (08) 114488
Artificial Intelligence: Applications to Logical Reasoning and Historical Research	K.S.Jones et al.	
Logic, Algebra and Databases	R.Ennals	
New Horizons in Educational Computing	P.H.D.Gray	
Microcomputers in Education	M.Yazdani	
Microcomputers in Education 2	C.Smith	
	E.Ramøden	

((PROLOG)(DATA))

Micro-PROLOG i skolen

Et interview med Jonathan Briggs

Ved Jens Rasmussen, DLH.

I bogen »Beginning micro-PROLOG« af Richard Ennals omtales forskningsprojektet »Logic as a Computer Language for Children«. Dette projekt havde til formål at undersøge mulighederne for at anvende micro-PROLOG i skoleundervisningen. Man interesserede sig dels for at afdække anvendelsesområder og dels for at udvikle en brugerflade, der kunne gøre det muligt for børn og nybegyndere at arbejde med Prolog. I denne omtale refereres en medforsker, Jonathan Briggs, for at have identificeret en række vanskeligheder, som viste sig, når børn arbejdede med Prolog. I april måned besøgte jeg Jonathan Briggs på Imperial College i London, for at få uddybet, hvori disse vanskeligheder bestod, og for at tale om, hvordan de kunne afhjælpes.

Briggs: I projektet »Logic as a Computer Language for Children«, som blev sat i gang af professor R. A. Kowalski med Richard Ennals som ledende forsker prøvede vi at se, om ideerne i Logic-programmering kunne bruges i skolen. Der er tale om noget ganske andet end at bruge ideerne til Prolog-programmering. Jeg mener, at man først må tage stilling til, om man vil undervise i Prolog eller i logisk programmering. Forskellen er i virkeligheden den, at logisk programmering er den teoretiske del, hvoraf Prolog kun er en begrænset version. Prolog-programmering er i visse henseender mere som ethvert andet programmeringssprog: Man kan skrive tydelige Prolog-programmer, hvis man vil det; men logisk programmering er langt mere ren og ægte.

Projektet har hele tiden beskæftiget sig med logisk programmering og ikke med Prolog programmering. Ideen i logisk programmering, hvor børn beskriver viden i meget ren forstand i regel- og factstermer, kan føre til stor forståelse af det udvalgte emne. Hvor computer-programmering i hvad sprog det end måtte være, er en færdighed i sig selv, så har projektet her altid været forbundet med det generelle aspekt.

JR: Hvad er det som først og fremmest adskiller micro-PROLOG fra Prolog?

Briggs: Micro-PROLOG blev i virkeligheden skrevet til vores projekt, og begrundelsen var, at forsker-teamet ønskede en prolog, som kunne anvendes i skolen, en som kunne køres på de maskiner, som findes i de britiske skoler. Prolog var udviklet til at køre på store mainframemaskiner; men micro-PROLOG blev skrevet til at køre på meget små maskiner. Derfor måtte der træffes en række

beslutninger, som kunne gøre det muligt. En af de beslutninger, som blev truffet, var, at en LISP-agtig syntaks skulle antages som et slags internt niveau, som et antal redskaber kunne bygges ovenpå. Det ville tillade børn ikke at skulle interagere med LISP-syntaksen; men i stedet bruge en langt mere »lige ud af landevejen«-syntaks. Den syntaks, som blev udviklet, blev kaldt SIMPLE.

JR: De vanskeligheder ved micro-PROLOG, du har gjort opmærksom på er altså vanskeligheder ved SIMPLE-versionen. Hvori består de?

Briggs: De ting, jeg specielt så på, da jeg startede, var nogle af de problemer, børnene havde med SIMPLE-syntaksen. Den blev designet meget hurtigt, og der var et antal vanskeligheder, jeg følte var blevet overset. For det første et par ret trivielle problemer:

a) Syntaksen er baseret på et lille antal nøgleord: add, which, does. Men i den første version af SIMPLE skulle der store bogstaver til at starte dem, og det betød, efter min mening, at der var indlagt en unødvendig vanskelighed. Brugeren var nødt til at tage stilling til, hvornår store bogstaver var nødvendige, og hvornår de ikke var.

b) For det andet var parenteser en meget vigtig del. Vi sætter parenteser om forskellige ting, og det ser ud som om vi har tre forskellige måder at anvende parenteser på, som alle var forskellige og forvirrende. Når man giver det til 9-12 årige børn, så kender de ikke den teoretiske begrundelse for, at det er sådan, for de kender ikke sproget.

Efter min mening er Prolog med SIMPLE et meget kraftfuldt redskab, så ideen var at tage denne Prolog med LISP-syntaks og tilføje et sæt redskaber, som kunne tillade børn at bruge den. Ved at tilføje disse redskaber forhindrede man i virkeligheden ikke børnene i at anvende den LISP-agtige syntaks, så problemet blev, at der i virkeligheden var nogle kommandoer, der, afhængigt af hvordan man taster dem, enten kunne virke med LISP-syntaksen eller SIMPLE-syntaksen. Hvis man f.eks. taster **list all** i små bogstaver får man SIMPLE listning; men hvis man taster i store bogstaver, fås den interne arbejdsinterframe. Der er gode grunde til, at det forholder sig sådan; men når vi prøvede det i virkeligheden, blev børnene forvirrede. Der var desuden en række andre ting, hvor SIMPLE-syntaksen efter min mening viste sig at være utilstrækkelig, så det blev min hovedopgave at skrive en anden syntaks. Den kom til at hedde MITSI, og den har »en manden på gaden«-brugerflade.

JR: Hvad karakteriserer MITSI-syntaksen?

Briggs: MITSI blev ikke skrevet til at udføre Prolog-programmering. Med SIMPLE kan man gøre næsten alle de ting, man også kan med Prolog, således at man kan skrive alle sine Prolog-programmer ved at anvende SIMPLE. MITSI blev skrevet, så det kun var muligt at gøre et vist antal af disse ting. Den er om du vil et »under-system«, alene beregnet til at introducere folk til Prolog, så de får fat i ideerne uden at skulle bekymre sig om syntaksen. Jeg følte ikke, at syntaksen var stridspunktet, så ved at opfinde en ny syntaks håbede jeg, at vi kunne få eleverne til hurtigere at forstå reglerne, facts'ene og måden at stille spørgsmål på.

MITSI er et lukket system. Man kan ikke foretage ting, som vil få effekt på den underliggende Prolog. Som sagt, hvis man i SIMPLE taster **list alle** eller **LIST ALL**, så har det to forskellige virkninger i samme system, hvilket jeg finder forvirrende. I MITSI-versionen hedder kommandoen **list all!** med et udråbstegn, for MITSI er tegnsætningsstyret. Man fortæller den, hvad man vil gøre ved at slutte med et tegn: punktum, udråbstegn, spørgsmålstegn. Hvis man f.eks. vil fortælle computeren at »John likes Mary«, så gøres det med et punktum: **Johna likes Mary.**

JR: Hvordan kan man indlægge information i MITSI-versionen?

Briggs: I SIMPLE og Prolog versionerne er der forskellige måder at indlægge information på:

John likes Mary	benævnes infix-syntaks.
John male	benævnes postfix-syntaks.
has (Mary the book)	benævnes prefix-syntaks.

I SIMPLE er alle disse syntakser mulige. I MITSI er kun infix-syntaksen mulig, og det er igen en forsimplicering for at gøre den indledende undervisning nemmere. Hvis man f.eks. vil sige »John male«, så gøres det på denne måde: **John sex male.** eller **John gender male.** eller hvordan man nu kan finde på at udtrykke det. Der er ingen tvetydigheder omkring syntaksen. Den er altid den samme. Som udvidelse heraf, den eneste tilladte udvidelse, kan disse argumenter være lister, hvilket giver mulighed for næsten hvad som helst med denne syntaks, blot man husker at der skal være tre elementer i en sentens.

JR: Hvordan skal spørgsmål formuleres?

Briggs: Jeg følte, at måden at stille spørgsmål på i SIMPLE, var for kompliceret. Den syntaks man var nødt til at lære så sådan ud:

which (x : John likes x)

Det er fint, når man ved, hvad det er man gør, og så er det ikke noget problem. Men som en begyndelse til at lære noget nyt, virker det som et chok. Der er for megen tegnsætning og for mange muligheder for at begå fejltagelser, så i MITSI vil det tilsvarende spørgsmål se sådan ud:

John likes someone?

Variablerne er markeret med ordet »some« på engelsk. Man kan sige **John likes someone?**, **John likes something?**, **John likes some-hvadsomhelst?**

»Some« markerer en hvilken som helst variabel. Jeg har altså prøvet at lave det uden parenteser, og det er den eneste tilladte måde at stille spørgsmål på i MITSI. I SIMPLE er der to slags spørgsmål. Der er »does«-spørgsmål og »which«-spørgsmål. »Does«-spørgsmål besvares med »yes« eller »no«. »Which«-spørgsmål giver svar. Hvis man i MITSI spørger: **John likes Mary?** svares der med »yes« eller »dont know«. Mange lærere klagede over at der blev svaret med »no«. De mente dette »no« var for stærkt. Derfor siger MITSI »dont know«. Systemet arbejder jo i en lukket verden, og computeren kan kun svare på det den ved. Derfor ændrede jeg »no« til »dont know«. Dette svar fås, hvis der ikke er nogen variabel. Er der variabel svarer den **Yes John likes Mary**, så den afgør hvordan den skal besvare spørgsmålet, afhængigt af om der er nogen variabel i spørgsmålet.

JR: Hvad er der ellers gjort for at »venliggøre« brugerfladen?

Briggs: Man kan, når et spørgsmål er blevet besvaret, spørge igen med »why«, hvorefter systemet kan give en forklaring ved at oplyse om de regler, det brugte til at afgive det første svar. Det skyldes, at vi arbejder med regler af formen »that/if« eller »that/at«, så hvis man spørger, **why?** er det sandt, kan computeren sige: »Det er sandt fordi det og det er sandt«. Det er meget nemt at skrive »why«, og det er en af de ting, som mangler i SIMPLE.

I det hele taget skulle disse ændringer gerne betyde, at de som begynder med MITSI er blevet ledt godt på vej til at gå videre med SIMPLE.

JR: Hvordan er disse resultater blevet fulgt op siden projektet sluttede for nu snart et år siden?

Briggs: Vi fortsætter med at prøve at gøre tingene nemmere for lærere. Vi er overvejende involveret i det, vi kalder at bygge »tool kits«. Vi bevæger os videre mod, hvad jeg ser som Prologs mest hensigtsmæssige anvendelsesform i skoler, nemlig ekspertsystemer. Vores interesse er ikke så meget at skrive black-box systemer. Jeg ønsker ikke at børn skal konsultere et ekspertsystem, det er ikke særligt interessant. Det vi koncentrerer os om, er ideerne om at lærere og elever i fællesskab skal involveres i at skrive et ekspertsystem. Begrundelsen herfor er især, at hvis eleverne præsenteres for et færdigt ekspertsystem, så kan det for dem være lige meget, om det er skrevet i BASIC, Prolog eller Assembler eller hvad det nu kan være. Men hvis de selv er involveret i processen at skrive et ekspertsystem, så vil de øjeblikkelig erkende, at det er ualmindeligt svært at skrive det i BASIC eller Assembler. Kan vi derimod designe det rette materiale, som børn kan bruge til at prøve at skrive ekspertsystemer i Prolog, så vil vi kunne gøre dem interesserede i disse ideer. For at antyde, hvad det er jeg mener, kan jeg nævne en pakke, som er på markedet. Den hedder »Apes«, og er beregnet på erhvervslivet. Det betyder, at den ikke kan køre på de maskiner, der almindeligvis er til rådighed på skolerne. Den fordrer maskiner i IBM PC-størrelse.

UNBEATABLE TOOLS FOR EXPERT SYSTEMS

LPA PROLOG + apes is an Expert System shell developed by Logic Based Systems Ltd in association with Logic Programming Associates to compliment their already successful Logic Programming languages.

apes — Augmented PROLOG for Expert Systems — is a suite of modules and by combining these modules a range of Logic Programming environments can be constructed, in particular, Logic Based Expert Systems.

KEY FEATURES

- Full documentation including tutorial examples describing how to build Expert Systems.
- Automatic generation of dialogue.
- Explanation of reasoning.
- Fully compatible with LPA micro-PROLOG and LPA sigma-PROLOG.
- Restrictive language syntax.
- Every module has been exhaustively tested in a number of applications in universities and industry.

APPLICATIONS

Applications have already been made in the following areas:—

- *Medicine
- *The Power Industry
- *Consumer advice
- *Financial marketing
- *Education

Selvom »Apes« ikke er designet til børn, arbejder vi alligevel lidt med den i skolen, og det har været meget spændende. Vi bruger den til et historiprojekt om sakserne. Vi vil udforske synet på saksernes historie. Vi prøver at få børnene til at gå ind i reglerne for, hvorfor et bestemt sted er egnet til boplads, ved at lade dem anbringe bopladser der, hvor der er gode forsvarsmuligheder. Hertil får de en liste af gode grunde til, hvorfor en anbringelse er god, og hvorfor en anden er dårlig. Der er f.eks. gode forsvarsmuligheder, hvis bopladsen anlægges på et højt beliggende sted. På mange måder former forløbet sig herefter som når man kører et Prolog-program. Der er noget information, men man vil komme ud for at bestemt information ikke er der, og så vil computeren i et Prolog-program komme tilbage og sige: **Dont know**; men »Apes« vil, hvis den ikke kender svaret, generere et spørgsmål til eleven, som hun kan besvare og derved lægge ny viden ind i systemet. »Apes« kan herefter afgøre om reglerne er gode nok: »Se, det var virkelig en dårlig placering af mange grunde. Hvorfor mener du sakserne valgte den? Hvad tror du fik dem til at vælge den på trods af den kendsgerning, at der er elendige forsvarsmuligheder?« Computersiden er her ikke særlig væsentlig. Det er snarere en måde at fokusere på ideer. Der er tale om en model, som konstrueres af læreren og eleverne.

JR: Sat på spidsen lyder påstanden om LOGO, at LOGO-programmering er godt for barnets hele udvikling og om Prolog lyder påstanden, at beskæftige med Prolog er godt for udviklingen af barnets logik. Hvad er din mening om sådanne omfattende forventninger?

Briggs: Jeg mener, vi kan forsyne børn og lærere med kraftfulde redskaber, som kan hjælpe dem med at tænke om eller på det, der interesserer dem. Det er ikke så afgørende om det er Prolog; men Prolog har givet mig en chance for at give lærere, hvad jeg håber bliver et kraftfuldt redskab. Det er et excellent sprog til at tilveje-

bringe »idémaskiner«. Det er nærmest sådan jeg opfatter Prolog. Jeg deler ikke nødvendigvis det synspunkt, som andre har givet udtryk for, at Prolog f.eks. skulle være den ny latin. (Det er en almindelig antagelse, at latins plads i rækken af skolens fag var begrundet i dets formaldannelsesmæssige egenskab. **JR**). Det er ikke det, jeg finder afgørende.

Jeg mener LOGO viser sin begrænsning, når man prøver at bruge den til kreativ skrivning. Jeg mener, det er langt vanskeligere at bruge LOGO til kreativ skrivning end Papert lader formode. Der er ikke så meget man kan lave. Man kan f.eks. ikke få en blomst til at åbne sig. Det ville være ualmindeligt svært at skrive i LOGO, fordi formen på tingene på skærmen hele tiden forandres. Der er dog stadig ingen, der har fundet en tilfredsstillende måde at lave Prolog-grafik på. Det skyldes især, at det kræver meget store maskiner; men det vil komme. Vi vil få det at se på PC'ere meget snart. Derimod er det nemt at implementere LOGO i Prolog; men det giver ikke Prolog-grafik.

MITSI-versionen er beskrevet i bogen: **Jonathan Briggs: »micro-PROLOG rules - an introduction to micro-PROLOG programming for the BBC-Micro, Research Machines 380Z, ZX Spectrum & other CPM & MSDOS Microcomputers**«, London 1984. Bogen er ikke kommercielt publiceret, men kan fås ved henvendelse til firmaet **Logic Programming Associates Ltd., 10 Burntwood Close, London SW18 3JU**. Tlf.: 00944 1 874 0350. Jonathan Briggs har været ansat i dette firma siden afslutningen på det omtalte projekt ved Imperial College.

Bogen findes nu også i en dansk oversættelse udvidet med micro-PROLOG og SIMPLEX. Den fås hos **Data-Prolog DK v/ Karsten Bøgh, Myntevej 3, 9380 Vestbjerg**. Tlf. 08 - 29 61 24.

Studieture

Studietur til England i efterårsferien 1985

Studiebesøget henlægges til London-området med besøg på skoler, andre uddannelsesinstitutioner, forskningscentre o.l.

Debatmøde med engelske kolleger om undervisningen i matematik og datalære. - Studieturen vil fokusere en del på datalære.

Herudover mulighed for sight-seeing og teaterture.

Turen arrangeres i samarbejde med vore engelske kontakter og henvender sig først og fremmest til undervisere i matematik og datalære (seminarieelever kan også deltage).

Varighed: 7 dage. Pris, incl. rejse (ab Esbjerg), hotel m. halvpension ca. 2250 kr.

Studietur til DDR i påskeferien 1986

Studiebesøget påtænkes henlagt til Berlin-området og vil omfatte besøg på folkeskoler, gymnasier, fagskoler, seminarier med henblik på at give en almen indsigt i undervisningssystemet i DDR i almindelighed, men med særlig vægt på undervisningen i matematik og evt. datalære.

Studieturen vil blive arrangeret i henhold til kulturaftalen mellem DDR og Danmark, og man kan derfor forvente, at den vil få et halvofficielt præg.

Studieturen henvender sig til undervisere i matematik og datalære (seminarieelever kan også deltage).

Varighed: 6 dage. Pris, incl. rejse (ab København), hotel m. halvpension, ca. 2100 kr.

Studieturene søges arrangeret med et deltagerantal på min. 20 pers., og interesserede bedes skriftligt henvende sig inden den 15. juli til:

Kaj Brandt Thomsen,
Kærbyvej 44,
8983 Gjerlev.

Tlf. 06 - 47 40 46 (privat) - 06 - 42 46 26 (skole).



MØLLEGADE 22
7430 IKAST
TLF. 07 - 15 47 11



**PICCOLINE
PARTNER**

BØGER
DISKETTER
PRINTERE
QUICK - SKEMALÆGNING
DASK SKOLEADMINISTRATION
SKØNSKRIFTSPRINTERE
TERMINALBORDE
ØKONOMISYSTEMER

RC-INFO

Information for RC-brugere

NYHEDER FRA PICCOLINEN

MF915 Kommunikationskort f. BBC/C64, 2 kanaler

Der kommer hele tiden nye ting til Piccolinen, og i sidste nummer af RclInfo blev tre nye interface-kort beskrevet. Det første af disse – »User port« kortet – er nu frigivet til salg, og prisen er holdt så lav som på 1650 kr. excl. moms.

Kortet har fået betegnelsen MF915 og indeholder to BBC-kompatible user porte, der er bygget op om kredsen 6522. Kortet monteres inden i Piccolinen i den generelle iSBX-fatning og ender i et 25 polet Cannon stik på bagsiden af Piccolinen. Hvis man ønsker det, kan man så ved hjælp af »adptoren MF919 få signalerne splittet op til to stik, der er identiske med det, der anvendes på BBC datamater. Så vil alt udstyr, der kan tilsluttes BBC datamaten også kunne tilsluttes direkte til Piccolinen – oven i købet i »stereo«. Yderligere en adaptor, der ender i et stik, som er identisk med Commodore64's user port kommer inden længe.

RC606P Matrix-printer

Her er så en »sværvægter« af en printer til Piccolinen. Faktisk er vagten på denne nye printer over 20 kg, så man kan ikke betegne den som transportvenlig. Men til gengæld er det andre faciliteter, der er lagt vægt på.

Udskrivningshastigheden er så høj som 240 tegn/sek., og der skrives med 24(!) nåle. Dette muliggør reel skønskrift, men så falder hastigheden så også til 80 tegn/sek., hvilket dog stadig er hurtigt i forhold til f.eks. typehjulsprintere. Printeren har i øvrigt 3 indbyggede tegnsæt, hvor de alle også kan skrives som kursiv skrift, og der er mulighed for programmerbare tegnsæt.

SW1407 RcFont

RcFont, de blev omtalt i forrige RclInfo, kan nu bestilles til en pris på 1285 kr. excl. moms.

Programmet giver mulighed for at definere sine egne tegn til såvel dataskærmen som RC603(604) printerne.

Dette åbner for flere spændende muligheder, f.eks. kan det græske tegnsæt, der findes i Piccolinens al-

ternative tegnsæt nu byttes ud med det russiske tegnsæt, og der kan så laves undervisningsprogrammer i russisk. Men man kunne også udnytte RcFont til at lave specielle symboler, der så kan danne ikoner, som udnyttes i et program.

Sammen med RcFont leveres også et hjælpeprogram, CHARSET.COMD, der lynhurtigt kan lægge et defineret tegnsæt ind i skærm eller printer. Brugeren af dette program i forbindelse med opstart af maskinen, kan man altså få Piccolinen til at starte op med kinesiske tegn – hvis man da har lavet et sådant alfabet . . .

Rødvin til (næsten) alle

Det er faktisk oftest de mange brugere af Piccolo og Piccoline, der finder de »fiduser«, der gør en udnyttelse af systemerne elegant eller nem.

Sådanne »fiduser« kan selvfølgelig være til nytte og inspiration for andre brugere, og derfor vil vi gerne formidle dem bl.a. via RclInfo og PICCOLINEN.

For at få gang i en sådan udveksling af »fiduser« til gavn for alle, sætter Regnecentralen gerne et par kasser rødvin (som selvfølgelig kan veksles til hvidvin) på højkant.

Reglerne er meget simple:

Hvis man har fundet frem til en programmeringsmæssig smart detalje, et utraditionelt anvendelsesområde eller lignende, sender man en beskrivelse heraf til Regnecentralen.

Hvis »fidusen« offentliggøres vil man modtage en eller flere flasker rødvin (eller hvidvin). Når vi foreholder os muligheden for at kunne give mere end én flaske, skyldes det, at vi forventer sådanne kreative og nyttige ideer, at vi simpelthen ikke vil kunne nøjes med en enkelt flaske. Antallet af flasker bestemmes egenmægtigt af »den gamle redacteur« og kan ikke appelleres.

Der gøres opmærksom på, at man ved indsendelse af »fidusen« gør denne til »offentlig ejendom«, som alle Piccolo- og Piccoline-brugere kan udnytte.

»Fiduserne« indsendes til:

Regnecentralen af 79
Lautrupbjerg 1, 2750 Ballerup
att.: RclInfo

To skråstreger med stor nytteværdi

Når der skal udføres CCP/M kommandoer, har der hele tiden været flere måder, dette kan gøres på.

Man kan selvfølgelig skrive kommandoerne direkte på tastaturet og får dem udført med det samme.

Men man kan også få dem udført automatisk ved opstart ved at skrive dem i en fil, der gemmes under navnet STARTUP.*n* (Hvor *n* står for konsol nummeret 0-3). Men her har man hidtil været begrænset til en enkelt kommando.

Endelig kan en samling af kommandoer samles i en submit fil, og de kan så udføres med kommandoen SUBMIT *filnavn*.SUB.

Kombinerer man de to sidste faciliteter, har man kunnet lave disketter, der automatisk sætte printeren til 2, indlæste grafikken og derefter startede RcComal80. Dette klares, ved at man laver en fil med følgende indhold:

```
PRINTER 2
GRAPHICS
COMAL80
```

Filen kan f.eks. få navnet START.SUB, og så skal man også lave filen STARTUP0, der skal indeholde kommandoen SUBMIT START.

Brugte man MENU systemet, havde man samme problem; skulle der udføres flere kommandoer efter hinanden, var man tvunget til at gå den indirekte vej via SUBMIT.

Alt det er jo en smule besværligt og det kræver også, at SUBMIT.CMD ligger på disketten o.s.v.

Men nu kan man klare den slags på en meget nemmere måde. Man kan nemlig samle flere kommandoer på samme linie, blot skal der sættes to skråstreger mellem de enkelte kommandoer. Dette betyder, at vil man lave en automatisk opstart af RcComal med grafik og printeren tilsluttet diskettestationen, skal man bare lave STARTUP0 med følgende indhold:

```
PRINTER 2 // GRAPHICS // COMAL80
```

Systemet vil så ved opstart udføre de enkelte kommandoer en efter en. Hvis man senere afbryder RcComal80, vil grafiksysteget stadig ligge i maskinens lager og fylde plads op. Hvis man tilføjer kommandoen GRAPHICS NO sidst i ovenstående række, vil denne blive udført, når RcComal80 forlades, og man vil så have »ryddet op« efter sig.

(I ovenstående eksempel bruges kommandoen PRINTER. Der burde måske have stået SKRIVER, da Piccolinen jo er dansk og da disse to kommandoer er identiske. Men da jeg aldrig personlig kan huske, om det nu hedder PRINTER eller SKRIVER, opfatter jeg det som en brugervenlighed i systemet, at det er lige meget hvad man skriver – eller printer . . . RcInfo, redaktøren).

Som nævnt er det en generel facilitet, og man kan derfor også skrive flere kommandoer direkte på tastaturet med skråstreger imellem, og man kan ligeledes bruge skråstregerne i MENU-systemet.

Har man brug for at få udført flere kommandoer, end der kan stå på en linie, må man stadig ty til SUBMIT, men den daglige brug lettes betydeligt ved denne nye facilitet.

OBS: Der er flere måder, man kan få de rigtige tekster ind i såvel STARTUP.*n* og submit filer.

Hvis man har RcTækst, kan denne umiddelbart bruges. RcTækst vil nemlig, når der sættes et punktum i filnavnet, undlade at gemme formatoplysninger m.v. i filen, der så vil være en ren tekstfil. (Dette er i øvrigt en facilitet, der også gør det muligt at udnytte RcTækst til redigering i f.eks. RcComal80 programmer, men herom en anden gang).

Man kan også lave disse filer ved hjælp af et lille RcComal80 program, der opretter filen med det rigtige navn og så simpelthen skriver indholdet derud med PRINT FILE.

Endelig kan man bruge PIP og skrive filen direkte fra tastaturet, som det også er beskrevet i installationsmanualen. Ønsker man at lave ovennævnte STARTUP0 fil, gøres det som følger:

```
PIP STARTUP0=CON: <return>
PRINTER 2 // GRAPHICS // COMAL80 // GRAPHICS NO <return> <ctrl j> <ctrl z>
```

PIP overfører tegnene fra tastaturet (CON:) til filen (STARTUP0) en for en, som de tages. Denne proces afsluttes med <ctrl z> Grunden til, at <ctrl j> skal med, er at et tryk på <return> ikke automatisk medfører linieskift i denne forbindelse. <ctrl j> er simpelthen koden for linieskift, og skal tilføjes efter <return> hver gang dette bruges.

CCP/M faciliteter

CCP/M har mange hjælpeprogrammer og indbyggede faciliteter, som man kan have nytte af. For en fuldstændig orientering om alle mulighederne henvises til den originale dokumentation fra Digital Research, der kan bestilles hos forhandlerne og Regnecentralen.

Et af de hjælpeprogrammer, som man kan have god nytte af, er SET. SET bruges til at sætte *fil attributter*, *password* og *automatisk registrering af tidspunkter for brug af filer*.

Formatet afhænger af den aktuelle opgave og kan f.eks. være:

```
SET Æ option Å
```

option betegner en af faciliteterne, som SET kan styre, *d*: er navnet på det diskette-drev, hvor den diskette, som man vil beskytte. *Filnavn* er navnet på en bestemt fil, som mærkes.

```
SET d: Æ option Å
```

```
SET filnavn Æ option Å
```

(Grunden til brugen af stort Æ og stort Å skyldes, at disse tegn svarer til firkantede parenteser i US-ASCII).

Der er ingen mulighed for her at komme ind på alle mulighederne med SET, men et par eksempler er:

Skrivebeskyttelse'

Hvis man ønsker at sikre en fil mod utilsigtet sletning eller ændring af indholdet, kan man give den attributten RO (Read Only):

SET *filnavn* Æ RO Å.

Man kan ændre tilbage igen til RW (Read Write) med:

SET *filnavn* Æ RW Å.

På samme måde kan en fil gøres til en system fil (SYS) eller en ikke-system fil (DIR):

SET *filnavn* Æ SYS Å og *filnavn* Æ DIR Å.

System filer vil ikke fremkomme ved DIR kommandoen, og man kan således »skjule« filer ved at sætte dem til SYS.

PASSWORD beskyttelse

Man kan knytte et password til selve password systemet på en diskette og til de enkelte filer. Når der først er knyttet et password til en fil, kan det specificeres, hvilken form for brug af filen, der kræver password. Man kan f.eks. lave det sådan, at alle kan læse i en fil, man at det kræver password at ændre eller slette filen.

Først skal disketten have et navn. Muligvis har den det i forvejen, f.eks. kan den have fået det ved initialiseringen. Man kan altså også bruge SET til at ændre navne på disketter:

SET *d:* Æ NAME= *diskettenavn* Å.

Man kan koble beskyttelsen ind og ud ved følgende kommandoer:

SET Æ PROTECT=ON Å og SET Æ PROTECT=OFF Å.

Men det vil jo ikke være nogen beskyttelse, hvis alle og enhver bare kunne koble hele systemet fra. Derfor kan der knyttes et password til selve diskettens navn, og har man først gjort det, kan man *kun foretage ændringer i password systemet, hvis man kender dette password*. Det er altså et password, som man skal huske på, for glemmer man det, kan man udelukke sig selv fra sine egne filer. Man sætter dette password med:

SET Æ PASSWORD=*password* Å (hvor *password* kan være på 8 tegn).

Det kan senere ændres eller fjernes, *men det kræver, at man selv kan huske det(!)*. Man fjerner det ved at give et tomt password:

SET Æ PASSWORD= <return> Å.

De enkelte filer tildeler man password sådan:

SET *filnavn* Æ PASSWORD=*password* Å.

Det er tilladt at bruge ? og ★ i filnavnet, så man kan med en operation sætte password på mange filer samtidig.

Typen af beskyttelse sættes på følgende måde:

SET *filnavn* Æ PROTECT=*type* Å.

Hvor *type* kan være READ, WRITE, DELETE og NONE.

Bruges READ kræves der password for alle operationer på filen. WRITE tillader, at der læses i filen, men alt andet kræver password. DELETE kræver password ved sletning eller ved ændring af filens navn, mens NONE fjerner password beskyttelsen.

Andre muligheder med SET

Piccolinen indeholder et ur og en kalender, som kan udnyttes på mange måder. Man kan bl.a. få systemet til automatisk at notere dato og tid for hvornår en fil er dannet og hvornår den sidst er opdateret eller brugt. Disse oplysninger får man frem, når man bruger SDIR til at udskrive kataloget.

Man kan koble denne mekanisme fra og til, men kan ikke have registrering af dannelsesestidspunktet og sidste brugstidspunkt samtidigt, da disse to muligheder alternerer.

Kommandoerne er:

SET Æ *type*=ON Å og SET Æ *type*=OFF Å.

Hvor *type* kan være: CREATE, ACCESS eller UPDATE.

Ved at sætte disse faciliteter i værk, kan man måske holde bedre orden på sine filer, der hele tiden bliver »datamærket«. Man vil også kunne bruge det i undervisningssituationer, hvor man kan få en slags registrering af aktiviteterne ad denne vej.

Vinduer og flere konsoller

(Eller lidt om fejltekster m.v.)

Piccolinen kan som bekendt udføre 4 programmer samtidigt, og ved hjælp af vinduer, kan man følge med i udskrifter fra de forskellige programmer.

I sidste nummer af RcInfo blev beskrevet, hvorledes man kunne starte RcComal80 i flere konsoller. Kombineres dette med vinduer, kan man f.eks. udnytte disse muligheder til større brugervenlighed. Nedenstående eksempel illustrerer bare en idé, der kan bygges videre på.

Fejltekster i almindelighed:

Når der opstår en fejl under programmering eller afvikling af et program i RcComal80, får man en fejlmeddelelse. Disse fejlmeddelelser kommer selvfølgelig på dansk og er gjort så klare som muligt – men er samtidig begrænset til få ord.

Man har mulighed for at ændre på den del af fejltekster, som kommer under kørsel af et program eller ved udførelse af kommandoer. På disketten ligger programmet GENERRM.CSV, der genererer en fil – COMAL80.ERM – med fejltekster, som indlæses af RcComal80 ved opstart. Man kan umiddelbart lave sine egne fejltekster ved at ændre på data-sætninger i dette program. Mangler filen med fejlmeddelelserne, vil RcComal80 i stedet for en fejltekst komme med fejls nummer, og man kan så finde teksten i manualen. Ved at undlade fejlteksterne bliver der i øvrigt mere plads til programmet og data. Det samme gælder også for Piccolo-versionen af RcComal80; her hedder programmet dog ERRMSG og datafilen hedder SYSTEXT.C80.

RcComal80 program til uddybende forklaringer

Ønsker man at give mulighed for en uddyggende forklaring på fejltteksterne direkte på skærmen, kan man udnytte muligheden for vinduer. Man kan f.eks. have startet RcComal80 i både konsol 0 og 1, og mens man programudvikler i konsol 0, kan man samtidig have et lille program (f.eks. som eksemplet) liggende fast i konsol 1, som når man indtaster et fejlnummer kommer med en fyldig forklaring.

I princippet behøver man egentlig ikke at bruge vinduer, men kan bare skifte mellem konsollerne ved hjælp af `<ctrl 1>` og `<ctrl 0>`.

Til tider kan det imidlertid være meget rart at have forklaringen på skærmen, samtidigt med man kan se på den del af programmet, som forårsagede fejlen, og dette kan klares med vinduer.

Man kan vælge mange måder at placere vinduerne i forhold til hinanden på, men den simpleste er nok, at lade de nederste 5 linier på skærmens rumme plads til forklaringen. I så fald skal vinduet for konsol 0 »skæres« af nederst på skærmen og kun gå til linie 18, mens vinduet i konsol 1 bare skal flyttes ned på skærmen, så øverste kant passer med underkanten af vinduet i konsol 0.

Dette gøres ved hjælp af VMENU eller VINDUE. For en nærmere beskrivelse af dette henvises til brugervejledningens kapitel 7, (som man absolut bør kikke i – der er mange muligheder for gode effekter!) men følgende kommandoer kan bruges og f.eks. lægges ind i de to STARTUP filer.

STARTUP0:

```
Vindue Ændres n=0 h=18 // graphics // comal80
```

STARTUP1:

```
Vindue Ændres n=1 r=20 //comal80 fejl1
```

Disse to startup filer vil installere vinduerne, starte RcComal80 og starte det lille fejlforklaringsprogram i konsol 1. (Bemærk, at man kun behøver at medtage de parametre, der skal ændres, i VINDUE kommandoen).

Hvordan det bruges . . .

Man skifter mellem de to konsoller ved `<ctrl 0>` og `<ctrl 1>`. Man kan på statuslinien se, hvilken konsol de er aktiv, og det man taster, vil alene berøre den aktive konsol. Opstår de nu en fejl under programmeringen, og man ønsker den uddybende forklaring, skifter man bare til konsol 1 og taster fejls referencenummer. Så finder det lille program teksten frem, og man kan så skifte tilbage til konsol 0 og gå videre. Hvis man ønsker at have hele skærmen til konsol 0 kan dette klares ved at trykke på A4 tasten. Den virker som en skiftekontakt, der aktiverer vinduet eller lader konsollen have hele skærmen.

Et praktisk råd . . .

Det er ikke alle fejlttekster, der har lige stort behov for uddybende forklaringer. Man kan derfor nøjes med at vælge dem ud, der erfaringsmæssigt volder problemer. Disse fejlttekster giver man så et referencenummer ved at rette i data-sætningerne i GENERRM.CSV. (Man kan f.eks. skifte stjernerne ud

med referencenummeret). Når så der kommer et nummer i stedet for stjerne, ved man, at der findes en uddybende forklaring til denne fejlttekst, og kan så skifte til konsol 1, hvis det ønskes.

Kommentarer til programeksemplerne

Disse skal alene betragtes som idé-oplæg til programmer af denne type. Fejltkiste ligger i en random fil, hvor record-nummeret bruges som referencenummer. FEJL2 kan bruges til at lægge teksterne ind i denne fil, og FEJL1 kan så senere hente dem og skrive dem på skærmen. Ved at ændre på variabelen »sidebredde« kan man få teksten udskrevet i forskellige bredder, hvilket kan være af interesse, hvis man vælger en anden placering og størrelse på vinduet.

God fornøjelse . . .

```
0010 // Programeksempel: FEJL1
```

```
0020 antalfejl = 2; sidebredde = 78
```

```
0030 DIM fejltkest$ OF 400
```

```
0040
```

```
0050 PROC hent__tekst
```

```
0060 OPEN FILE 1,"TEKSTER", RANDOM 400
```

```
0070 READ FILE 1,fejlnr.: fejltkest$
```

```
0080 CLOSE
```

```
0090 ENDPROC hent__tekst
```

```
0100
```

```
0110 PROC vis__tekst
```

```
0120 PRINT CHR$(31);
```

```
0130 liniebrud = 1; sidste__tegn = LEN  
(fejltkest$)
```

```
0140 WHILE liniebrud + sidebredde(sidste__tegn  
DO
```

```
0150 søg = liniebrud + sidebredde
```

```
0160 WHILE fejltkest$(søg) < > "" DO søg =  
søg+1
```

```
0170 PRINT fejltkest$(liniebrud:søg)
```

```
0180 liniebrud = søg + 1
```

```
0190 ENDWHILE
```

```
0200 PRINT fejltkest$(liniebrud:sidste__tegn)
```

```
0210 ENDPROC vis__tekst
```

```
0220
```

```
0230 PRINT CHR$(12)
```

```
0240 REPEAT
```

```
0250 fejlnr.: = 0
```

```
0260 WHILE fejlnr < 1 OR fejlnr > antalfejl DO
```

```
0270 PRINT AT(1,1,CHR$(30))
```

```
0280 INPUT AT(1,1,"Skriv fejlnummer: ": fejlnr
```

```
0290 ENDWHILE
```

```
0300 hent__tekst
```

```
0310 vis__tekst
```

```
0320 UNTIL FALSE
```

```
0010 // Programeksempel: FEJL2
```

```
0020 DIM linie$ OF 80,fejltkest$ OF 400
```

```
0030 CREATE "tekster",2
```

```
0040 OPEN FILE 1,"tekster", RANDOM 400
```

```
0050 WHILE NOT EOD DO
```

```
0060 READ nr.,linie$
```

```
0070 WHILE linie$ < > "q" DO
```

```
0080 fejltkest$ = fejltkest$ + linie$ + ""
```

```

0090 READ linie$
0100 ENDWHILE
0110 WRITE FILE 1,nr: fejltekst$
0120 fejltekst$: = ""
0130 ENDWHILE
0140 CLOSE
0150
0160 DATA 1,"DISKETTEN ER SKRIVE-
      BESKYTTET:"
0170 DATA "Man kan ikke skrive på en beskyttet"
0180 DATA "diskette, dvs. man kan hverken
      skrive"
0190 DATA "i datafiler eller gemme programmer"
0200 DATA "Skrivebeskyttelsen kan fjernes fra"
0210 DATA "disketten, hvis det ønskes.
      Se herom"
0220 DATA "i Brugervejledningen afsnit 2.4.1."
0230 DATA "e"
0240 DATA 2,"ARITMETISK OVERLØB:"
0250 DATA "RcComal80 har et bestemt
      talområde,"
0260 DATA "og kommer man uden for det,
      vil man"
0270 DATA "få aritmetisk overløb – eller
      underløb,"
0280 DATA "som der også kan være tale om.
      Grænserne"
0290 DATA "er både for negative og positive tal
      fra"
0300 DATA "1.000...E-128 til 9.999...E + 126."
0310 DATA "e"
0320 // o.s.v. ....
    
```

CCP/M 3.1 – oversigt over distributionsdisketterne

CCP/M 3.1 release 2.3 til Piccolinen består af 3 disketter, der tilsammen indeholder omkring 140 filer. Det vil være ganske få, der nogensinde vil få brug for alle filer, men har man valgt at medtage rub og stub på distributionsdisketterne i stedet for at skulle sende forskellige disketter ud.

Langt de fleste vil kun få brug for ganske få af de mange filer, og takket være menu-systemet behøver man egentlig ikke at sætte sig ind i hvilke filer, der bruges til hvad.

Men alligevel kan det være rart at have et overblik over indholdet på disketterne – om ikke for andet, så til tilfredsstillelse af nysgerrigheden.

Diskette 1/3 indeholder de filer, som man oftest har brug for, og langt hen af vejen er det kun diskette 1/3, man skal have fat i.

Diskette 2/3 indeholder filer, som bruges hvis man selv vil generere CCP/M systemer og netværks systemer samt hjælpefiler, der skal bruges, hvis man kører med lokalnet.

Diskette 3/3 indeholder en række standardiserede CCP/M systemer, der nok dækker langt de flestes behov, og der vil derfor kun i særlige tilfælde være behov for at generere systemer ved hjælp af diskette 2/3.

Diskette 1/3:

Filer, der har tilknytning til MENU-systemet:

Disk vedl. CMD. Diskettevedligeholdelse, dvs. kopiering, initialisering, verificering.

Filadm. CMD. Filadministration, dvs. oversigt, kopiering, sletning, omdøbning.

Function CMD. Programmering af funktionstaster.

Gkonfig. CMD. Konfigurering af grafiksysteem, dvs. valg af drivere.

Imenu. MDF. Installationsmenu.

Instal. CMD. Installationsprogram.

Konfig. CMD. Konfigurering af porte, skærm, ur, disksystem, lokalnet mv.

Lanrel. CMD. Testprogram for lokalnet.

Menu. CMD. Styreprogram for menuer.

Menu. MDF. Standard hovedmenu.

Menu vedl. CMD. Vedligeholdelse af menuer.

Okonfig. CMD. Konfigurering af opstart-fil (KDEF.SYS).

Siorel. CMD. Testprogram for seriel port (V24-port).

SYS vedl. CMD. Systemvedligeholdelse, dvs. test af skriver, diskstation, lokalnet, V24-port.

Filer, der har tilknytning til GRAFIK:

Assign. SYS. Tekstfil med navne på de aktuelle drivere.

Assign. 1. DIR. Mulige drivere for et 1-konsol-system (bruges af Gkonfig.).

Assign. 4. DIR. Mulige drivere for et 4-konsol-system (bruges af Gkonfig.).

DD604H. SYS. RC604 Printer A4-bred.

DD604HC. SYS. RC604 Printer A4-bred farve.

DD604V. SYS. RC604 Printer A4-høj (samt RC603).

DD604VC. SYS. RC604 Printer A4-høj farve.

DD621. SYS. RC621 Plotter.

DD759H1. SYS. Skærm, høj opl., 1 konsol.

DD759H4. SYS. Skærm, høj opl., 1 konsol, digitizer.

DD759HD4. SYS. Skærm, høj opl., 4 konsoller, digitizer.

DD759HM1. SYS. Skærm, høj opl., 1 konsol, mus.

DD759HM4. SYS. Skærm, høj opl., 4 konsoller, mus.

DD759M1. SYS. Skærm, farve, 1 konsol.

DD759M4. SYS. Skærm, farve, 4 konsoller.

DD759MD1. SYS. Skærm, farve, 1 konsol, digitizer.

DD759MD4. SYS. Skærm, farve, 4 konsoller, digitizer.

DD759MM1. SYS. Skærm, farve, 1 konsol, mus.

DD759MM4. SYS. Skærm, farve, 4 konsoller.

Graphics.CDM. Installation af grafiksysteem.

Gkonfig.CMD. Konfigurering af grafiksysteem.

CCP/M filer:

ASM86.CMD. Assembler for intel 8086.
CCPM. SYS. CCP/M-systemfilen med styresystemet, indlæses med opstart.
CHSET.CMD. Sæt attributter på programfiler.
DATE.CMD. Udskriv dato og tid.
DDT86.CMD. »Kig« i fil.
DIR.CMD. Oversigt over disketteindhold.
ED.CMD. Linieorienteret editor.
ERA.CMD. Sletning af fil(er).
ERAQ.CMD. Sletning med bekræftelse.
GenCMD. Generering af CMD-fil fra Hex.
HELP.CMD. Udskrivning af hjælpepetekster.
HELP.HLP. Hjælpetekst-fil.
INITDIR.CMD. Ændring af diskettekatalog.
PIP.CMD. Kopiering af fil(er).
PRINT.CMD. Udskrivning af filer på printer.
REN.CMD. Omdøbning af fil.
SDIR.CMD. Oversigt over disketteindhold.
SET.CMD. Sæt attributter på filer.
SHOW.CMD. Vis oplysninger om disk.
SUBMIT.CMD. Automatisk udførelse af kommandoer.
SYSTAT.CMD. Systemstatus, dvs. bl.a. oversigt over systemets ressourcer.
TYPE.CMD. Udskrivning af fil på skærm.
VCMODE.CMD. Tilstand for virtuel konsol.
VINDUE.CMD. Vise/ændre vindues-definition.
VMENU.CMD. Installering af vinduesstyring.

Diverse:
COMAL80.CMD. Comal80-systemet.
COMAL80.ERM. Comal80 fejlmeddelelser.
FILEX.A86 Overføringsprog. fra Piccolo.
FILEXL.CMD. Filex.A86 = kildekode.
FRIGIV.CMD. Frigiv disk eller printer.
GENERRM.CSV. Comal80-prog. som opretter Comal80.ERM.

INST1JOB.SUB Installation af
INSTJOB.SUB Comal80-diskette/system-diskette.

KDEF.SYS. Konfiguration til indlæsning ved opstart.
NYCOMAL.SUB. Kommandofiler til fornyelse
NYSYSTEM.SUB. af system på gl. diskette.
RESERVER.CMD. Reserver disk eller printer.
START0. Opstartfiler for
START80. Comal80-diskette/systemdiskette.
STARTUP.0. Opstartfil for disk 1/3.

Diskette 3/3:

Standard CCP/M systemer:

CCPM1.SYS. System 1 konsol uden net.
CCPM1R.SYS. System 1 konsol netbruger.
CCPM1RS.SYS. System 1 konsol netbruger + vært.
CCPM1S.SYS. System 1 konsol netværk.
CCPM4.SYS. System 4 konsoller uden net.
CCPM4R.SYS. System 4 konsoller netbruger.
CCPM4RS.SYS. System 4 konsoller netbruger + vært.
CCPM4S.SYS. System 4 konsoller netværk.

Diverse:

CCPM.MDF. Menu til valg af system.
CCPM.SYS. System 1 konsol uden net.
MENU.CMD.
ERA.CMD. Som på disk 1/3.
PIP.CMD.
SET.CMD.
STARTUP.0. Opstartfil for disk 3/3.

Ønsker man at udnytte de mange muligheder, som er tilgængelige i det avancerede CCP/M system, var det måske en god idé at anskaffe nogle af de supplerende manualer, der kan fås til Piccolinen. Disse manualer er på engelsk, og der kan f.eks. være tale om:

SW1498D Piccoline Programmer Guide.
 SW1696D DR-NET Manual Set (Digital Research).
 SW1698D GSX-86 Manual Set (Digital Research).
 SW1699D CCP/M-86 Manual Set (Digital Research).

Regnecentralen

LAUTRUPBJERG 1
 2750 BALLERUP
 TLF. 02 - 65 80 00

Hvordan kursus i datalære?

Af Klaus Nørskov, Roskilde.

Mange kommuner rundt om i landet afholder i denne tid kurser for deres lærere i datalære/informatik. Lige så mange steder gør man sig overvejelser vedrørende målet, indholdet og omfanget af disse kurser.

Jeg vil her skitsere, hvordan man prøvede at afhjælpe de mest akutte problemer på dette område i Roskilde kommune.

I slutningen af 1983 blev der på fælleslærerrådets foranledning nedsat et EDB-udvalg i Roskilde kommune, som havde til opgave: »At undersøge og komme med forslag til, hvordan de muligheder der er indeholdt i den ny teknologi, kan udnyttes i undervisningen ud fra pædagogiske og etiske synsvinkler«.

Udvalget bestod af fem lærere, som i fællesskab repræsenterede et bredt udsnit af de forskellige klassetrin og fag i folkeskolen.

Udvalget udarbejdede en rapport til skolekommissionen. Rapporten indeholdt, udover betragtninger vedrørende pædagogik og informatik, et forslag om afholdelse af et 60 timers kursus i datalære/EDB. Kurset blev bevilget, og kursusansøgerne blev udvalgt, så alle skoler var repræsenteret med 2-3 lærere.

Vi var to af udvalgets medlemmer, der skulle forestå kurset. Det var fra starten udgangspunktet, at kurset **ikke** kunne træde istedet for årskurserne i datalære/informatik på DLH.

Som det fremgår af nedenstående kursusopslag, var det ikke tanken at »kvalificere« kursisterne til at undervise i datalære.

Kurset havde det mål at orientere kursisterne om både datalære, datamatstøttet undervisning, datamatformidlet undervisning og ikke mindst de etiske, filosofiske og pædagogiske/psykologiske aspekter, der knytter sig til brugen af datamater i undervisningen. Ca. halvdelen af tiden blev brugt til programmering og maskinens opbygning/virkemåde.

Den resterende del af tiden blev brugt til at diskutere datalærens plads i folkeskolen (mål, indhold, klassetrin etc.), samt informatikkens placering i de enkelte fag. Spørgsmål såsom: Hvad er vores holdning til datamatformidlet undervisning? hvilke krav må vi som lærere stille til eventuelle programmer? hvilken indflydelse har datamaskinen på vores opfattelse af virkeligheden? har informatik og samtdsorientering noget med hinanden at gøre? gav mange gode og frugtbare diskussioner.

Kurset var opbygget således at formiddagene alminde-

ligvis, på baggrund af konkrete oplæg, blev brugt til gruppearbejde og diskussion. Gruppearbejdet tog typisk udgangspunkt i forskellige artikler eller i vurdering af undervisningsmaterialer/undervisningsprogrammer. Gruppearbejdet afsluttedes med en plenumdiskussion. Om eftermiddagene arbejdede kursisterne normalt med programmeringsøvelser i grupper af 2-4 personer. Undervisningen var tilrettelagt sådan at grupperne havde mulighed for at arbejde i individuelt tempo.

Resultatet af kurset blev ikke at Roskilde kommune fik 30 »nye« lærere til at undervise i datalære. Man kan ikke lære at programmere og sætte sig ind i de didaktiske og metodiske problemstillinger, der knytter sig til datalære på 60 timer. Resultatet af kurset blev, at man har sikret sig, at der på samtlige skoler i kommunen findes et antal lærere, der kan bruge en datamaskine, og som har indsigt i dens pædagogiske muligheder og begrænsninger.

OPSLAG

8-7130-1. Kort kursus i datalære.

Kursussted: Amtscentralen, Østergade, Roskilde.

Kursusomfang: 1. del. Mandag den 18/6 – torsdag den 21/6 kl. 9-16.30.

2. del: Medio oktober 1984 (1 uge).

Formålet med kurset er at give deltagerne kendskab til datamatens muligheder og begrænsninger gennem beskrivelse af databehandlingens vigtigste anvendelsesområder og deres samfundsmæssige fordele og ulemper.

Kursets indhold rummer en gennemgang af databegrebet, programmeringssprogene COMAL, LOGO og trafik STS, datamaskinens opbygning (CPU, hukommelse/lager, ydere enheder), didaktiske og metodiske overvejelser i relation til datalære, datamaskinstøttet undervisning og dataformidlet undervisning. Endvidere en omtale af undervisningsmaterialer, EDB i undervisningen samt etiske, filosofiske og pædagogisk-psykologiske overvejelser i tilknytning til informatikkens anvendelse i undervisningen.

Der sigtes på at den sidste kursusuge koncentrerer om en deltagerstyret erfaringsudveksling på grundlag af konkrete oplæg.

Der kræves ingen særlige forudsætninger, og det tilstræbes ved optagelsen, at alle skoler i kommunen får repræsentanter på holdet.

Boganmeldelser

Lokalnet - hvad er det?

Indføring og case for projektdeltagere.
Dian Kjærgaard og Bo Svanborg.
Forlaget Samfundslitteratur, København.
224 sider, kr. 80,00.

Denne bog tager udgangspunkt i datamater og databehandling på et introducerende niveau. Derefter behandles datatransmission, som munder ud i bogens egentlige ærinde, nemlig at gennemgå datanet og lokalnet. Endelig slutter bogen med krav til lokalnet samt vurdering af tekniske muligheder i en konkret organisation.

Formålet er ifølge forfatterne Dian Kjærgaard og Bo Svanborg at give en tilstrækkelig grad af indsigt i de såkaldte informationsteknologier til, at en læser

- kan være med i en debat om indførelse af ny teknologi på arbejdspladsen eller i samfundet.
- kan diskutere med teknikere og leverandører i den situation, hvor konkret anskaffelse af teknologi overvejes og
- får et grundlag, som gør det lettere at tilegne sig mere dybtgående teknisk kundskab om ønsket.

Bogen kan medvirke til, at flere brugere får kendskab til, hvordan datanet og lokalnet fungerer, således at de ikke er overladt til edb-teknikerens ensidige påvisning af teknikkens ufejlbarligheder.

Bogen er let læselig med en klar opdeling af de forskellige nettyper med gode illustrationer, som kan give ideer til pædagogisk tilrettelæggelse af et vanskeligt emne. En række centrale begreber forklares og sættes ind i en sammenhæng.

Er man eller skal man som bruger involveres i edb-systemarbejde, kan man næppe finde fremstillinger, der rammer mere i plet end denne.

A. Egebjerg Johansen.

Skomal

Viggo Sadolin.
Teknisk Forlag A/S.
43 sider, 49 kr.

Bogen giver en indføring i brug af skomal-fortolkeren, d.v.s. at man skal have en diskette med denne fortolker og en dertil passende datamat.

I forordet nævnes en række maskiner, som skulle have

skomal. Desværre har det ikke været muligt for mig at få skomal til den maskine, som jeg har - et ikke ukendt fænomen, som forlaget nok burde være mere opmærksom på.

Skomal (skolens matematiske algoritmesprog) tilstræber at give eleverne samme inspirerende arbejdsform som logo, idet de med skomal kan eksperimentere med algoritmer, variable og funktioner.

Skomal viser os en lagerside og en normalside. På lagersiden kan vi indlæse tildelinger, ordre og værdier, og maskinen viser et billede af, hvordan vi kan forestille os lagerset i en datamat se ud. På normalsiden kan vi indskrive programmer og få dem udført uden adgang til at »se ind« i maskinen.

Procedurer og parametre kan illustreres på fineste måde, det gælder såvel afleverende som modtagende parametre, og programmer kan afvikles trinvis, så man let kan følge processerne og evt. finde fejl.

Desværre har jeg ikke kunnet prøve skomal selv eller på elever, men efter at jeg så skomal første gang for snart 2 år siden, lavede jeg et lille comal-program, som efterlignede lagersiden i simpel funktion, og dette har jeg haft en del glæde af i arbejdet med elever. Så jeg mener, at skomal kan tilføre undervisningen i regneprocesser væsentlige og spændende elementer.

Jens Krog.

EDB i undervisningen

Red. af Karl Johan Jørgensen og Poul Svendsen.
Billesø & Baltzer.
238 sider, 164 kr.

Bogen er opdelt i en teoridel og en praksisdel.

I teoridelen har en række mennesker med indsigt og engagement i problematikken omkring indførelse af edb/informatik i skolen skrevet indlæg, hvor emnet belyses fra flere sider, og forskellige holdninger kommer til udtryk. Jeg vil illustrere dette med nogle citater fra denne del af bogen:

Om LOGO: »Brugt på denne måde er datamaskinen en af de bedst tænkelige udfordringer til en divergerende tænkning, som vist er langt vigtigere for vore børn i den verden, de skal leve i, end den konvergerende tænkning«.

»Lad os erkende at informatik mere er et intellektuelt univers af kvalitet, end et materielt univers af kvantitet«.

Praksisdelen rummer dels en række artikler af folk, som har erfaring med brug af edb i forskellige fag og dels korte beskrivelser af praktiske forsøg udført i folkeskolen. Jeg vil også her bringe nogle citater fra artiklerne: »Det er ikke et krav til en dialog til undervisningsbrug, at den skal være et effektivt problemløsningsværktøj, og det er vel netop en væsentlig forskel i forhold til de data-sprog, som benyttes i erhvervslivet«.

»Den fortsatte vekselvirkning mellem maskinen og mennesket giver et dynamisk samspil, hvor hver af parterne styrker hinanden. Maskinen bidrager med stor regnehastighed og den overskuelige grafiske afbildning af resultaterne - mennesket bidrager med forståelse, overblik og målrettet indsigt«.

Ovenstående er kun en lille smagsprøve - resten er også i høj grad værd at læse.

Afsnittene om forsøgene er skrevet af de implicerede lærere og er en fortræffelig oversigt over de væsentlige erfaringer, som er gjort og som de, der vil arbejde med edb i folkeskolen, kan have megen nytte af at kende.

Jeg finder bogen meget inspirerende og værende en god opslagsbog i gjorte erfaringer. Den bør være tilgængelig for alle de, der i de kommende år skal tage fat på indførelse af informatik i folkeskolen.

Jens Krog.

CP/M for Begyndere.

Murtha, S. og Waite, M.
Borgen 1985.
201 s. Kr. 175.

Af og til diskuteres det, hvem der har begået den mest ulæselige dokumentation for et system. Afgørelsen er ikke helt klar, men den originale vejledning for CP/M er mellem kandidaterne.

Dette har medført, at der udgives et stort antal bøger, der søger at råde bod på Digital Research's uformåenhed. Den foreliggende bog er en af disse udgivelser.

Bogen, der kun omhandler det, der nu kaldes CP/M-80, rummer 9 kapitler og to tillæg, og der er en indholdsfortegnelse og et register. Først forklares CP/M's opbygning, systemstart behandles og formattering og kopiering gennemgås. I forbindelse med kopiering forklares brug af PIP, og også STAT omtales. Editoren ED gennemgås ret omhyggeligt, og samme behandling får DDT og ASM. Sidste kapitel tager sig af SUBMIT og XSUB. I de to tillæg vises CP/M's struktur samt en oversigt over faste kommandoer og hjælpeprogrammer.

Bogen er helt tydeligt lettere at forstå end den officielle dokumentation, men den behandler til gengæld ikke alle dele af CP/M. Forfatterne undgår heller ikke for korte forklaringer, men som helhed er bogen af en sådan karakter, at den vil kunne være en hjælp for den kun lidt øvede CP/M-bruger.

Fritz G. Knudsen.

Programmering SÅDAN.

Ole Monrad.
Teknisk Forlag.
263 sider, 178 kr.

Skal man forstå bogens titel ultimativt? Det mener forfatteren ikke, selvom der er meget med:

Problemløsning: Takkebrev.
Tekstbehandlingsystem.
Indlæringsprogram.
Søjlediagram m.v.
Metodebeskrivelse: Klammediagrammer, (Warnier-Orr.)
Pseudokode.
Omsætning fra pseudokode til BASIC,
COMAL-80 og PASCAL.
Behandling af skærm og tastatur.

I et appendix vises forskellige strukturers udseende ud fra ovennævnte metodebeskrivelser. Feks.:

GENTAGELSE (EN ELLER FLERE GANGE) (REPEAT).

1) Klammediagram:	begin	
	(1)	
	program	udfør handling x
	gentagelses-	
	struktur	.end
		(1)
2) Pseudokode:	GENTAG	
	programdel	
	INDTIL betingelse.	
3) BASIC:	100 REM gentag det følgende	
	200 programdel	
	300 IF NOT betingelse THEN	
	GOTO 100.	
4) COMAL-80:	100 REPEAT	
	200 programdel	
	300 UNTIL betingelse.	
5) PASCAL:	REPEAT	
	programdel;	
	UNTIL betingelse;	

Til slut en kommenteret litteraturoversigt og et godt index.

Jeg har lidt svært ved at vide hvem og til hvad, jeg skal anbefale denne bog, da jeg ikke har brugt den i praksis. Den indeholder som antydte mange elementer med forslag til opgaver, alt beskrevet i et let og inspirerende sprog. Så jeg mener, den må have interesse for datalærere til selvstudium og til emnet problemløsning.

-bpp-

Turbo Pascal version 2.10A, CP/M-85.

Pascal er som bekendt et sprog, der har sit udspring og sin første udvikling i undervisningskredse, her forstået som universitetskredse. Dette har haft flere fordele til følge for Pascal som undervisningssprog, fordele, der vel egentlig i nogen grad kan genfindes i Comal.

Til tider har den stringens, der præger Wirth og Jensens definition, dog hindret praktisk brug af sproget, og der er derfor opstået versioner, der forsøger at kombinere de to verdener. Nogle Pascal-purister har sammenlignet sådanne versioner med den professionelle idrætsmand, der af og til anvender et »udvidet« regelsæt for at opnå resultater, men trods dette er Compas Pascal – et typisk eksempel – vel den mest udbredte Pascal-version på mikrodata-maskiner i Danmark.

Turbo Pascal er som Compas en professionel Pascal til brug på mikrodatamaskine. Turbo er en næsten komplet udførelse af Wirth og Jensen definitionen, men dertil kommer et stort antal udvidelser som random access filer, overlays, strenge, program chaining, operativsystemkald, absolutte adresser for variable og meget mere. Specielt kan nævnes in-line maskinkode, der giver mulighed for optimere særligt udvalgte programstumper.

Turbo er en integreret editor/compiler, der foruden at have en meget stærk editor også oversætter lynhurtigt. Eksempelvis kan det nævnes, at et programkompleks, der

findes i 5 filer med ialt godt 3000 linier pascal-tekst oversættes til en ★.CMD-fil på disketten i løbet af godt to minutter! Afviklingen af de oversatte programmer er på samme måde meget hurtig.

Er Turbo og Compas ens?

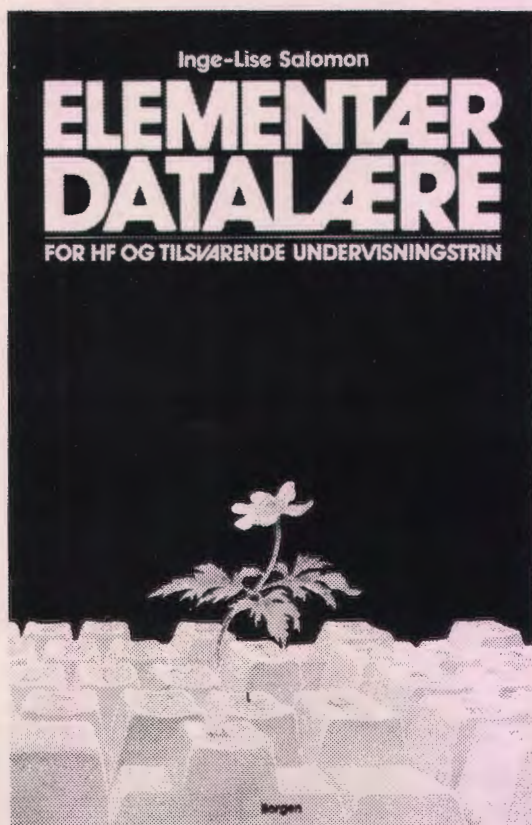
For Compas-brugeren må det efterhånden være klart, at Turbo ligner Compas en del. Der er små forskelle: Man trykker kun på en kommandotast, hvor Compas også kræver return, det hedder **length** i Turbo, **len** i Compas, og der er flere småting, der nok ikke ændrer programstrukturen, men alligevel vanskeliggør en flytning af programmer. Lighederne er imidlertid så mange, at man fristes til at formode, at der er tale om samme ophavsmand til de to systemprogrammer.

Det er der nok ikke til dokumentationen, hvor Turbo kommer med en bog på næsten 300 sider, der er let læst, fyldt med eksempler og – meget brugbart – forsynet med et godt register. Bogen er naturligvis på engelsk, men den er et godt eksempel på velskrevet dokumentation.

Der medfølger et regnearksprogram, der er velegnet til videre bearbejdning, men der savnes muligheder for at benytte f.eks. GSX-grafik.

Turbo Pascal er således et godt alternativ, hvis man står overfor at skulle anskaffe en Pascal til brug på mikrodata-maskine.

Fritz G. Knudsen.



Nu 3. oplag

Inge-Lise Salomon
**Elementær datalære for HF
og tilsvarende
undervisningstrin**
304 s. illustreret. Kr. 124.

Fortæller om edb-maskiner, anvendelse af edb, mennesker og edb, samt programmering i COMAL. En grundig bog, der er velegnet til aftenskole og selvstudium, samt undervisning i gymnasiet.

Borgen

Efteruddannelse af lærere på Vestlolland i informatik

Af Jens Krog, Nakskov.

Allerede i skoleåret 81/82 enedes man i de 4 vestlollandske kommuner om en fælles indsats for at efteruddanne lærere, så de kunne klare den kommende undervisning i informatik.

Man startede med nogle korte (9 lektioner) introduktionskurser, hvor datamaten og nogle anvendelsesprogrammer blev introduceret. Ud fra dette kunne deltagerne så vælge fortsætterkurser med hovedvægten lagt på enten praktisk brug af maskinen og færdige programmer eller et lidt længere kursusforløb om programmering. Der var ikke nogen deltagergrænse: Alle, der ønskede det, kom på et af kurserne, og de, der var særligt interesserede, kunne vælge begge kurser. Holdstørrelsen var fastlagt ud fra antallet af maskiner, således at der var to kursister om hver maskine.

Samtidigt arbejdede man på at få DLH engageret i efter-

uddannelsen, og det resulterede i informatikkursus 1 både i skoleåret 83/84 og i 84/85. I skoleåret 85/86 bliver der informatikkursus 2 både i Nakskov og i Maribo.

Alle disse kursusaktiviteter har betydet, at datamater i flere år har været anvendt på de vestlollandske skoler og at der på praktisk talt alle skoler (også de mange 7-klassede) er lærere, der har en grundlæggende viden om edb i skolen, og på alle overbygningsskolerne er der lærere, som kan starte undervisning i datalære på valghold.

I Nakskov er man indstillet på til efteråret at give alle lærere fri en dag om ugen (»rullende« over ugens 5 dage) fra kl. 11 til kl. 14, så de kan deltage i det landsdækkende lærerkursus om skole-informatik.

Alt dette har givet os en rolig og fornuftig udvikling støttet af forudseende ledere og politikere.



NÆSTE NUMMER

Indlæg til næste nummer skal være redaktøren i hænde senest tirsdag 3. september.

Almindelige oplysninger om foreningen

Folkeskolefraktionen

FRITZ KNUDSEN

Kollerupvej 17, 8900 Randers
tlf. (06) 43 49 04

Indmeldelse i fraktionen kan ske til kassereren

BO BOISEN PEDERSEN

Hjarupvej 7, 6200 Åbenrå
tlf. (04) 62 79 60

Gymnasiefraktionen

JØRGEN F. HANSEN

Midgårdsvvej 7, 8600 Silkeborg
tlf. (06) 81 24 47

Indmeldelse til fraktionen kan ske til kassereren

POVL HOLM

Kildevej 16 B, 3300 Frederiksværk
tlf. (02) 12 48 59

Handelsskolefraktionen

LISBETH ANDERSEN

Århus Købmandsskole

Vester Allé 8

Postboks 308, 8100 Århus C

tlf. (06) 12 91 88.

Kontaktpersoner til øvrige områder:

Teknisk skole-området

KNUD SNOER

Gelballevej 69, 6640 Lunderskov

tlf. (05) 58 52 83

Andre undervisningsformer

KNUD ERIK KRISTENSEN

Myntevej 39, 8240 Risskov

tlf. (06) 17 77 08

BLADET:

Ansvarhavende redaktør:

FLEMMING HOLT

Drosselvej 21, V. Hassing

9310 Vodskov

tlf. (08) 25 71 47



**Kan du
se det
for dig?**

Jet 80 er en Z80-baseret datamat med CP/M plus (3.0) styresystem.

Fordele ved Jet 80 er:

- stort internt lager, 128 Kbyte RAM –
- stor diskettekapacitet, 2×800 Kbyte –
- stort programudvalg, alt til CP/M –
- hurtig og sikker med stort tilbehørsprog –
- stik til Europakort Bus for tilkobling af alle signaler.

HØJ YDELSE TIL LAVPRIS!

Jet 80 kan fås i mange modeller, f.eks.:

Jet 80 G – grafikmodellen med det flim-

merfri, monokrome 640×400 punkts billede (se ill.). Tekst og grafik kan vises hver for sig eller samtidigt på den lækre 14" skærm.

Jet 80 F – farvegrafik med 512×512 punkter i 16 forskellige farver, naturligvis inklusive drivprogrammer.

Jet-Net – hvor op til 30 datamater og 3 fælles printere kan sammenkobles i et lokalnet under BRIDOS. Alle Jet 80 typer kan anvendes.

FÅ YDERLIGERE OPLYSNINGER HOS:

SC **METRIC** AIS

DATAAFDELINGEN • SKODSBORGVEJ 305 • 2850 NÆRUM • TELF. 02-804200

Piccoline kan, hvad der kræves — nu og i fremtiden

PICCOLINE hviler ikke på laurbærrene

PICCOLINE er nu solgt i knap 5000 eksemplarer, og succesen følges op med nye faciliteter og udvidelser, som gør PICCOLINE til den danske skolemikro — både nu og i fremtiden.

Regnecentralen er og har i mange år været den største leverandør af mikroer til undervisningssektoren, og RC tilbyder udstyr, der lever op til tidens krav, og som er kompatibelt såvel fremad som bagud.

PICCOLINE er en ægte 16 bit mikro med et standard styresystem, og PICCOLINE kan indgå i et lokalnetssystem på lige fod med RC's Partner mikrodatamat.



30 år med dansk edb
Regnecentralen

Lautrupbjerg 1
2750 Ballerup
02-65 80 00

Nye moduler og programmer til PICCOLINE:

BBC user port

Denne facilitet giver PICCOLINE mulighed for tilkobling af ydre enheder, som er konstrueret til BBC-datamater. Det drejer sig om robotter, pegetastatur osv.

Proceskort

Dette kort er beregnet til at PICCOLINE kan styre og måle med forskelligt udstyr. Det kan håndtere stepmotorer, relæer, kontakter og signalkonvertering.

RcFont

RcFont er et program, hvormed man kan definere sit eget tegnsæt, f.eks. græske og russiske bogstaver. Systemet kan også anvendes til at skrive store tegn til f.eks. overheads. RcFont udskrives på RC603 eller RC604 skrivere.

PICCOLINE vil til stadighed blive opdateret med de faciliteter, der skal til, og det gælder også på programmelsiden.

JA TAK

Send mig mere information

Navn: _____
Adresse: _____
Telefon: _____