

1

5. ÅRGANG

NOVEMBER 1980

data lære

INDHOLD

Klimaforværring

Cirkulæreskrivelse om datalære

COMAL 80 - hvorfor og hvordan

BAKSPOR

SOFIE

Generalforsamling

Udgivet af

FORENINGEN FOR DATALÆRE OG ANVENDELSE AF EDB I UNDERVISNINGEN

ISSN 0107-0002

Derfor bør De vælge CBM til undervisning - se især det femte punkt!

● Attraktiv pris:
komplet anlæg (32 k brugerdisponibelt
lager) med disktestation og printer
kr. 37.240 excl. moms.

● Comal 80 (foråret 81), Pascal, Basic.

● Danske vejledninger
- særdeles instruktive.

● Systemer for mange
anvendelsesområder
- administration
- planlægning
- tekstbehandling
- ingeniørberegninger
- processtyring

● - og sensationen:

De kan nu sammenkoble op til 8 CBM
maskiner med fælles printer og med
læsning/skrivning i de samme registre
placeret på disktestation.
Kapacitet: 2 x 175 k eller 2 x 512 k.
Investeringen er beskeden. De slipper
for de langsomme kassettebånd - og
sammenlignet med minicomputer får
De flere undervisningspladser for de
samme penge.



Ingeniørskolen, Horsens
Undervisning i EDB, byggetekniske beregninger
i forbindelse med kursusarbejde.

CBM

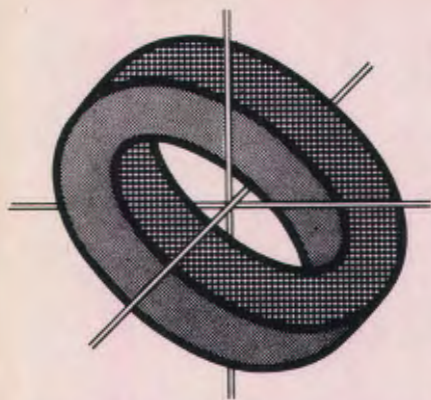
- den hårdeste af de
»små hårde kendsgerninger«!

Ring på (05) 61 11 00
og få nærmere oplysninger!

INSTRUTEK

CBM-computere fra Commodore, USA
- en af verdens førende microcomputerproducenter -
markedsføres i Danmark af Instrutek,
der har hovedkontor i Horsens, (05) 61 11 00
og kontor i København, (01) 41 34 00.

Klimaforværring



Direktoratet for gymnasieskolerne og HF har udsendt et nyt cirkulære om datalære. Direktoratet ønsker at fremme forsøg med datamaskinens anvendelse i den daglige undervisning i fagene samfundsfag, geografi, biologi, kemi, fysik og matematik, og det fremgår, at "det er direktoratets opfattelse at fremtidige retningslinier for disse fag kommer til at indeholde bestemmelser om, at der i undervisningen skal indgå anvendelser af datamatiske metoder". Direktoratet ønsker altså at bringe ovennævnte fag ajour med hensyn til anvendelse af edb, og det er naturligvis *positivt*, men cirkulæreskrivelsen indeholder der ud over vurderinger og påstande, der må give ethvert medlem af Datalæreforeningens gymnasiefraktion kuldegysninger, og som efter endt læsning efterlader det spørgsmål, hvad det egentlig er, der trænger sig på i direktoratet?

Direktoratet argumenterer for, at datalære ikke skal indføres i gymnasiet ved at henvise til fagtrængslen og den opsplittede skoledag. Der skal ikke indføres *endnu* et lille fag.

For det første: Hvorfor er datalære nødvendigvis et lille fag? – Hvad forstår direktoratet iøvrigt ved et lille fag?

For det andet: Forsøg under den eksisterende gymnasiestruktur kan finde sted på den måde, at datalære indgår *i stedet for* et af de eksisterende fag. Skoledagen bliver derved ikke mere opsplittet - tværtimod vil tilstedeværelsen af datalære, som direktoratet helt korrekt skriver, påvirke sammenhængen i positiv retning.

For det tredje: Direktoratet argumenterer for afvikling af datalære som selvstændigt fag ved at henvise til fagtrængslen og den opsplittede skoledag. Det er altså forhold inden for skolens verden, der henvises til. Hvad med samfundets krav til edb-undervisning i skolen? Direktoratet afstikker fremtidige udviklingslinier for datalære, men burde *det* ikke være en politisk afgørelse?

Det antydes i cirkulæreskrivelsen, at datalære i overgangsfasen fortsat kan være et selvstændigt fag - men endemålet? Sammenlignet med forholdene i Norge og Sverige (DATALÆRE nr. 4, september 1980) ser det ikke behageligt ud. Hverken i en faglig eller i en samfundsmæssig sammenhæng.

Det forekommer ikke rimeligt at antage, at faget uden en fast forankring som selvstændigt fag på timeplanen, kan fastholde det fagligt-pædagogiske niveau, der er forudsætningen for at fagets *tværfaglighed og metodik* (direktoratets ord) kan komme andre fag og integrationsprojekter til gavn.

Ligeledes kan man frygte, at de ressourcer, der er etableret i form af maskinelt udstyr m.v., vil vise sig ikke at kunne udnyttes i rimeligt forhold til den investerede kapital.

Det er også tvivlsomt om "de meget entusiastiske lærere der arbejder med faget" fortsat vil kunne holde gejsten oppe.

Endelig kan den nødvendige *efteruddannelse* ikke tilvejebringes uden faste rammer og ressourcer - et problem man netop i Norge har fået øjnene op for. Direktoratet skriver i cirkulæret: "Det kræver ikke mange timers selvstændigt arbejde før en lærer kan anvende maskinen i den daglige undervisning i de ovennævnte fag". Det forekommer os, at der med dette cirkulære er lagt op til en udvikling, der meget nemt kan resultere i datalæres reduktion til "lommeregnerniveau" i andre fag.

Skrivelsen omtaler "de meget entusiastiske" lærere, der har arbejdet med faget, og som i fremtiden skal være med til at give faget sin profil. Direktoratet bedes ved lejlighed uddybe, hvad der menes med dette. Hvad er det for et fag, der omtales, og hvad vil det sige, at give det en profil? Nærlæser med cirkulæreskrivelsen, synes det at fremgå, at direktoratet sætter faget datalære lig med summen af de eksisterende fags maskinanvendelser. Det fremgår nemlig, at HF-enkeltfagskurser får lov til at fortsætte forsøgsundervisningen i faget datalære med den begrundelse, at en enkeltfagslev ifølge sagens natur ikke kan samle flere fags edb-elementer op.

Som man ser, har cirkulæreskrivelsen bevæget sig ud på dybt vand, og det er svært at se, hvilken planke underskriveren, undervisningsinspektør Ib Fischer Hansen, egentlig vil klamre eller har klamret sig til. Det er i hvert fald ikke Datalæreforeningen. Cirkulæreskrivelsen har ikke været forelagt foreningen til udtalelse. Fantasien har frit spillerum for gætteri. Har man konsulteret de institutioner, der uddanner gymnasielærere i faget datalære? Næppe, for så ville Datalæreforeningen blive orienteret indirekte. Med arbejdslivets organisationer, der beskæftiger sig med edb-uddannelse? Nej, for i så fald kunne cirkulæreskrivelsen næppe tage afstand fra "entusiastiske elever med dybtgående kendskab til fagets tekniske muligheder". Har man skævet til rapporter fra Sverige, Norge eller England? Uden tvivl, ja, men i så fald har man læst hurtigt hen over de bilag, der fortæller hvilke institutioner og fagfolk, der andre steder har været involveret i så væsentligt planlægningsarbejde.

Kort og godt må man nu kræve, at direktoratet ophører med at politisere gennem cirkulæremidiet og at direktoratet åbner sig demokratisk udadtil som en offentlig planlægningsinstitution. I begge tilfælde: Om igen, Ib Fischer Hansen.

Arbejdsgruppen

Cirkulæreskrivelse om datalære

fra

UNDERVISNINGSMINISTERIET

DIREKTORATET FOR GYMNASIESKOLERNE OG HØJERE FORBEREDELSESEKSAMEN

Faget datalære har nu eksisteret som forsøgsfag i 8 år, og direktoratet finder tiden moden til en vurdering af resultaterne og en beslutning om de fremtidige udviklingslinier. Efter direktoratets opfattelse bør udviklingen nærme sig en integration af datalære i skolens øvrige fag gennem en gradvis afvikling over en årrække som selvstændigt fag på timeplanen.

Ved forsøget med datalære og ved en lang række andre forsøg i gymnasiet og på højere forberedelseksamen har det vist sig at et af de største problemer er fagtrængslen. Et andet tilbagevendende problem har været opsplitningen i mange små fag der har gjort det vanskeligt at opnå den fornødne tværfaglige sammenhæng i uddannelsen. Det er derfor ikke formålstjenligt at indføre endnu et lille fag i uddannelserne. Den tværfaglige og de metodiske færdigheder som datalære giver, vil kunne opnås ved en integration af fagets metoder i andre fag, samtidig med at uddannelsernes sammenhæng påvirkes i positiv retning.

Direktoratet finder det meget værdifuldt at datalære i opbygningsfasen fungerer som selvstændigt fag. De meget entusiastiske lærere der arbejder med faget, har været og skal i fremtiden være med til at give faget sin profil. Specielt har det vist sig at man i projektdelen og i specialerne har opnået resultater med henblik på at åbne skolen udadtil og komme i kontakt med mange arbejdspladser såvel inden for det offentlige som det private arbejdsmarked. Disse aktiviteter bør naturligvis bevares i den fremtidige placering af datalærens metoder og færdigheder inden for andre rammer.

De skoler og kurser som med stor succes har gennemført forsøgsundervisning i datalære, opfordres nu til at koncentrere forsøgsundervisningen omkring integration af datalære i fagene samfundsfag, geografi, biologi, kemi, fysik og matematik. Det er direktoratets opfattelse at fremtidige retningslinier for disse fag kommer til at indeholde bestemmelser om at der i undervisningen skal indgå anvendelser af datamatiske metoder.

Ca. 2/3 af skolerne har nu anskaffet en eller anden form for datamatisk kapacitet, og derfor kan

størstedelen af eleverne forlade uddannelserne med et grundlæggende kendskab til datalærens metoder og en vis færdighed i at anvende disse metoder. Den foreslåede ændring af forsøgsaktiviteten over en årrække sigter mod i højere grad at indføre mange elever i datalærens grundlæggende begreber frem for at give nogle få entusiastiske elever et dybtgående kendskab til dens tekniske muligheder.

Efteruddannelsen af lærerne i de ovennævnte fag tænkes ikke systematisk at blive optaget i efteruddannelsesprogrammet. De anvendelser der sigtes på, kræver ikke en længere efteruddannelse af lærerne, men vil kunne opnås gennem selvstudium på de skoler der råder over maskinkapacitet. Det kræver ikke mange timers selvstændigt arbejde før en lærer kan anvende maskinen i den daglige undervisning i de ovennævnte fag. Det er dog af største vigtighed at der produceres forskelligt materiale gennem forsøgsundervisning og gennem faglige efteruddannelseskurser som kan benyttes i undervisningen. Direktoratet opfordrer derfor til at skoler eller kurser som har forsøgsundervisning, rundsender materiale til de øvrige skoler og kurser. Samtidig vil direktoratet opfordre de faglige udvalg til at optage datalærens metoder i de fremtidige faglige kurser i de nævnte fag. Denne opfordring er allerede givet gennem flere år gennem forhandlinger med pædagogisk samarbejdsudvalg.

Vedligeholdelsen og tilsynet med maskinerne er endnu et uløst problem. Det hænger sammen med tilsvarende spørgsmål for sproglaboratorier, AV-udstyr, kopiudstyr, trykkeudstyr og så videre. Det ligger udenfor direktoratets muligheder at finde en løsning herpå, da forhandlingerne herom ikke foregår i direktoratets regi. En løsning synes ikke at være umiddelbart forestående, blandt andet på grund af den økonomiske situation.

For at give de tidligere årgange en mulighed for at studere datalære tænkes faget bevaret på enkeltfagskurserne, men udvidet til et 8 points fag i overensstemmelse med de ønsker der er fremkommet i forsøgsrapporterne. Det vil ikke være rimeligt at ældre studerende skal gennem flere fag for at opnå datalærens grundbegreber.

Ib Fischer Hansen/ Stig Obel
uv. insp.

Indlæg til cirkulæreskrivelsen

Den af ministeriet pr. 3. 10. 1980 udsendte cirkulæreskrivelse, der er gengivet på side 4, har sat sindene i bevægelse også uden for gymnasielærernes egne rækker (se lederen).

I Politiken fra 16. 10. 1980 skrev William Cauchi under overskriften: "Datalære afskaffes som fag i gymnasiet" følgende:

Efter otte års vellykket forsøgsundervisning i det nye fag datalære, som både elever og erhvervs-liv har vist stigende interesse, har Direktoratet for Gymnasieskolen nu overraskende udsendt et cirkulære, hvori det bekendtgøres, at datalære ikke længere er et selvstændigt fag.

Af hensyn til den almindelige fagtrængsel og et ønske om større tværfaglighed skal datalære i fremtiden integreres i samfundsfag, geografi, biologi, kemi, fysik og matematik.

Men lærere i disse fag skal ingen undervisning have i brugen af datamaskiner og datamatisk systematik. Det er nok med nogle få timers selvstudium, hævdes det i cirkulæret.

– Det er faktisk datalærens afskaffelse i gymnasieskolen, siger datalogi-professor H. B. Hansen, Roskilde Universitets Center (RUC). Det er et slag i ansigtet på de tvæsnede lærere, der har udført en virkelig pionergerning på dette vigtige felt. Og det er en helt fantastisk afstandtagen fra "Johnsen-rapporten", den undervisningsministerielle udredning i sin tid. Den sagde klart og tydeligt, at datalære bør indpasses i alle skoler på alle trin - naturligvis af lærere, der er uddannede til det.

Det er ikke realistisk at tro, at en hvilken som helst geografilærer eller kemilærer uden anden data-uddannelse end et kortvarigt selvstudium kan påtage sig at indføre gymnasieelever i datafagets metoder og grundliggende begreber, sådan som cirkulæret forlanger det af dem.

Enten er cirkulæret udtryk for en politisk vilje til total aflivning af datalæren – eller også er det baseret på en forbløffende mangel på kendskab

til, hvad der kræves for at kunne undervise forsvarligt i dette fremtidsfag, slutter professor H. B. Hansen.

Datalæreforeningen har til sine medlemmer udsendt en tilsvarende kommentar, og der kan også ventes protester fra anden side.

I samme avis havde professor Peter Naur, Københavns Universitet, dette indlæg under "Lærerne mener":

Ifølge et nyt cirkulære fra Undervisningsministeriet skal undervisningen i datalære i gymnasiet varetages af lærere i samfundsfag, geografi, biologi, kemi, fysik og matematik uden at disse har mere baggrund for datalære end få timers selvstudium.

Det svimler for tanken, hvad er det dog der foregår derinde i ministeriet?

Mener de det alvorligt? Mener de at vi skal vende tilbage til fortidens skole, hvor enhver falleret student kunne blive lærer?

Eller mener de at faget datalære ikke rummer noget indhold der kræver systematisk studium for at beherskes?

Mener de således at de sidste års drøftelser af mangelen på kompetente dataloger og datamatikere i både offentlige og private virksomheder er det rene gas?

Eller føler de sig således politisk presset, måske af de traditionelt veletablerede fags organisationer, at de er parat til at prøve at bedrage offentligheden ved at fortælle at datalæren indgår i gymnasiets undervisning, uanset at man er klar over at faget ikke støttes af kompetente lærerkræfter?

Man vægrer sig ved at tro at undervisningsministeren kan mene dette cirkulære alvorligt, det må være en skrubtudse. Lad det hurtigst blive trukket tilbage og erstattet af et som ministeren kan være bekendt.

Desuden har vi modtaget nedenstående indlæg fra Carsten Hansen, Greve:

Direktoratet ønsker ikke datalære i gymnasiet!

Når dette læses, er cirkulæreskrivelsen vedrørende datalære ca. en måned gammel. Jeg går derfor ud fra, at indholdet af den vil være bekendt for interesserede læsere af DATALÆRE, også uden for gymnasielærernes rækker. Der er imidlertid ikke tvivl om, at det vil kræve en meget omfattende debat at få direktoratet til blot at

overveje at ændre signaler, hvorfor også andre end de mest direkte involverede (såsom arbejdsgruppen for gymnasiet og HF og de lærere, der har haft mulighed for at deltage i forsøgsundervisning, samt de få dataloger, der har ansættelse i gymnasiet) må søges inddraget i debatten.

Nedenstående synspunkter skal også ses i lyset heraf; efter min mening er "slagtningen" af datalære som selvstændigt fag en helt konsekvent følge af direktoratets visioner om den fremtidige undervisning først og fremmest i gymnasiet. HF spiller i denne forbindelse en mindre fremtrædende rolle (som sædvanlig!).

Datalære - et fag?

Med den seneste udvikling in mente må det erkendes, at Datalæreforeningen og dens medlemmer har haft et naivt forhold til besvarelsen af ovenstående spørgsmål. Der er næppe mange af os, der ikke opfattede den "lille" gymnasie-reform for et par år siden som en "blåstempling" af datalære som fag i gymnasiet; at indførelsen skulle ske på bekostning af et hårdt trængt humanistisk fag, var imidlertid en teknokratisk "smart" løsning på et timeplaceringsproblem, som det må undre, om direktoratet kunne tro kunne ske uden sværdslag - og det ønskelige i en sådan udskiftning må være problematisk langt ind i vor forenings rækker. Det er let at være bagklog, men var det virkelig meningen at indføre datalære, eller var der tale om et paradeforslag? Det må vi kræve klart svar på nu, hvor udgangen er blevet den modsatte af den, der for meget kort tid siden var så godt som officiel direktoratspolitik.

Overskriften på dette afsnit var også titlen på et regional mødeforedrag af fagkonsulenten for fysik. Det er ikke ukendt, at fagområdet elektronik - på trods af stigende anvendelser heraf i arbejdslivet (tidligere hed det erhvervslivet) og stor betydning for mange "mellem" uddannelser - ikke har kunnet finde plads i gymnasiets fysikundervisning, når bortses fra ubetydelige speciale-anvendelser. Det var derfor ikke uden visse bange anelser, jeg afventede oplysninger om, i hvilken retning, indholdet af foredraget gik; det kunne jo være en af direktoratets kendte prøveballoner. Det krævede ikke de store kombinationsevner at sammenholde dette med de planer for det forsøg med fremtidens gymnasium på Herlev Statsskole, som direktoratet har stigende besvær med at bortforklare nyder særligt fremme der. En hovedide bag forsøget er integration af fagene, hvilket i sagens natur gør det mindre ønskeligt at øge udbudet af fag i gymnasiet.

Afgørelsen var derfor trods alt nærliggende.

Man forstår godt, at direktoratet efter en forsøgsperiode på 8 år følte det påtrængende med en afgørelse af datalæres fremtidige status i gymnasiet; alligevel er det et spørgsmål, om der på nuværende tidspunkt er tilstrækkeligt materiale til at træffe en så definitiv afgørelse, som det nu er sket. Direktoratet har ikke i særlig grad ønsket at fremme, at universitetsuddannede dataloger fra starten deltog i forsøgsarbejdet, og det har naturligvis også været væsentligt, at dette arbejde blev forestået af lærere med en pædagogisk erfaring, som de nyuddannede dataloger ikke kunne have. Dette har imidlertid betydet, at forsøgene i de første år foregik på en faglig baggrund, som for lærerens vedkommende var svagere end ved "normal" undervisning, hvilket utvivlsomt har haft betydning for fagets udformning i de første år, og som helt sikkert har betydet, at data-

læres karakter af selvstændigt fag ikke er kommet frem i tilstrækkelig grad i de første år. Dette ikke være sagt for på nogen måde at nedvurdere den pionerindsats, der blev ydet - blot må man gøre sig klart, at forudsætningerne for forsøgene var problematiske; de økonomiske forudsætninger gjorde også deres til at begrænse de faglige muligheder, ligesom direktoratets lunkne holdning til at fremme anskaffelse af baggrundslager og mere end een arbejdsplads, som vel nu anses for uomgængeligt nødvendigt, hvis man vil lave andet end talknuserprogrammer med små datamængder.

Nu, hvor man kan foretage materielindkøbene til en helt anden pris og få en helt anden kvalitet, og hvor et betydeligt antal af "nye" og "gamle" pionerer selv har sørget for en overordentlig kvalificeret efteruddannelse (ubetalt af arbejdsgiveren!), kommer så direktoratets nedlæggelse af datalære som selvstændigt fag. Så vidt jeg har fået oplyst, foreligger der i de nyeste forsøgsrapporter (velsagtens udarbejdet i sommerferien og igen ubetalt af arbejdsgiveren!) forslag til videreudvikling af faget, som er i bedste overensstemmelse med, hvad direktoratet finder værdifuldt, men som forudsætter datalære som selvstændigt fag. Har det haft indflydelse på cirkulæreskrivelsens udformning - og hvorfor ikke?

Kort sagt forekommer det undertegnede, at beslutningen er taget på et mangelfuldt grundlag, og at perioden på 8 år må være af mindre betydning i det foreliggende tilfælde.

Det fremgår af cirkulæreskrivelsen, at direktoratet ønsker undervisning i faget datalære i det fremtidige gymnasium, og at der ikke er tale om et særligt lille fag, fremgår af dets omfang på enkeltfags-HF. Det må imidlertid anses for at være nyskabende, at et fag af det omfang ikke kan tilstås timer på skemaet, og det må nødvendigvis få konsekvenser for ikke mindst bevillingsmæssige forhold, som jo er afgørende for de ydre rammer, som ikke mindst for datalære er af afgørende betydning. Og så er der spørgsmålet om lærernes kompetence. . .

Om kompetence

Efter at have reduceret datalære til et rent anvendelsesfag placeret under allerede eksisterende fag, præciserer direktoratet på en uvant beundringsværdig måde, hvad der kræves for at få undervisningskompetence i faget. Når man erindrer sig den besynderlige fremgangsmåde, direktoratet benyttede i en periode i forbindelse med oldtidskundskab (der var det igen!), hvor kompetence ikke var veldefineret, kan man konstatere, at direktoratet har taget ved lære: En lærer kan forestå undervisningen efter få timers (udokumenteret!) selvstudium. Sådan!

Fremgangsmåden er absolut ikke uden perspektiver. Nu, hvor der skal tilbydes datalære som valgfag i folkeskolens 8. og 9. klasse, kan

man forudse, at læreren kan få svært ved af nå elevernes niveau med den nævnte efteruddannelse. Hvem der skal undervise hvem, må da stå hen i det uvisse, men det skal måske forstås som direktoratets bud på, hvordan problemet med den faldende realløn for gymnasielærerne (som siden 1973 har mistet 25 % på livslønnen i realløn) skal løses: I de timer, hvor der skal undervises i "data-lære", nøjes læreren med at føre protokollen.

Man må desuden spørge sig, hvem der i gymnasieundervisningen skal kunne påtage sig at orientere om de samfundsmæssige følger af EDB-teknikken i offentligt og privat regi, eksempelvis hvad angår teknologivurdering og anvendelse som automatisering og processtyring (måske ønsker man slet ikke sådan undervisning?). De perspektivløse og derfor uinteressante anvendelser, som direktoratet lægger op til med de (lærer)ressourcer, man vil stille til rådighed, berettiger næppe (efter min mening ikke) til at tale om undervisning i datalære.

Direktoratets demokratiske sindelag

Et meget vigtigt overordnet formål for faget datalære er at sikre eleverne (som for de gymnasiale uddannelser overstiger 50 % af en årgang en del steder og er stigende på landsplan) en forståelse for de muligheder, anvendelse af EDB åbner for det moderne samfund. Ikke mindst anvendelser i offentligt regi giver eksempler på forskellige former for misbrug eller tilløb til misbrug af teknikken. Fagets etik indtager herved en central position; teknikken kan ret ubemærket medføre centraliseringer, som er til fare for demokratiet. Der er ikke tvivl om, at der uanset gymnasiets holdning til datalære vil ske den (særdeles målrettede!) uddannelse af arbejdskraft, der er nødvendig for de erhvervsmæssige anvendelser (såvel offentlige som private). Den form for talent, der kræves, gør det sandsynligt, at rekrutteringen i vid udstrækning vil ske fra gymnasiets potentielle klientel. Såfremt man overhovedet har tiltro til de gymnasiale uddannelsers almindelige karakter (som er nedfældet som overordnet mål i bekendtgørelsen), må man også tillægge det værdi, at undervisning i datalære sker i sammenhæng hermed.

En udlægning af det tidligere omtalte forsøg på Herlev Statsskole er, at man søger at opprioritere de humanistiske og samfundsvidenskabelige fag i forhold til de naturvidenskabelige på matematisk linie; ved anskaffelsen af de højeste niveauer i matematik og fysik og indrettelse af dem som redskabsfag for visse af de øvrige løber man den risiko, at de bliver orienteringsfag, hvor mangel på basisviden kan medføre, at det bliver læreren (eksperten!), der løser problemerne og præsenterer sin løsning. Det synes jeg er en u hensigtsmæssig måde at efterligne det "virkelige" liv på;

eksempler fra energidebatten belyser, hvad en rimelig basisviden kan betyde for den enkelte borgers mulighed for at vurdere eksperterts udsagn (vækstmodeller, f. eks. eksponentiel vækst, som nu er indført i matematikundervisningen i sprogligt gymnasium). Trods alt er det meget naivt at tro, at alle eksperter er uafhængige; de fleste er afhængige af både deres arbejde og deres løn, og det kan tænkes, at det påvirker deres indlæg i debatten.

Man savner præcisering af, hvorledes direktoratet ønsker sådanne tendenser modvirket for vort fagområde. (En nærliggende mulighed ville være indførelse af datalære som selvstændigt fag!). De hidtidige erfaringer fra en del skoler har mere end antydnet, at manglende eller utilstrækkelige muligheder for kvalificeret undervisning i datalære (mangelfulde og for små anlæg, lærerkompetence og allokerede lærerressourcer) bevirker, at det bliver få og særligt interesserede elever, der får et dybtgående kendskab til de tekniske muligheder - for det meste ved selvstudium. Det er mere end svært at se, at der automatisk ved anskaffelse af datalære som selvstændigt fag vil kunne ske en ændring heraf; disse få elever har fungeret - og, må man frygte - vil fortsat fungere som eksperterne, og datamaten vil blot være en intelligent skrivemaskine for det store flertal.

Vort ubetalte merarbejde

I en del år har de fleste af Datalæreforeningens lærere med tilknytning til gymnasieskolen af forskellige (gode) grunde udført gratisarbejde i forbindelse med opbygningen af faget datalære, i strid med GL's generelle henstilling om at lade være. Nu, hvor direktoratet har fejlet de væsentligste begrundelser for trods alt at udføre arbejdet af bordet, må det være på tide at tage honorar-spørgsmålet op til fornyet overvejelse. At direktoratet påstår, at forhandlingerne herom ikke foregår i direktoratets regi, er simpelt hen for billigt. Det er klart, at direktoratet må gøre forhandlerne fra Finansministeriet opmærksom på, at der er et område, der kræver betalt arbejdskraft - det vil næppe være uden virkning.

★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★

● ● ●

Datalære - forsøgsbeskrivelser

Artiklen i sidste nr. af DATALÆRE om datalære og forsøgsundervisning opfordrede til omhu med forsøgs-rapporteringen til brug for de omtalte katalogbeskrivelser. I denne forbindelse henvises der til en kommende skrivelse herom. Den er nu kommet (21. august 1980) og kan findes i Regelsamlingen, RS 01-19, eller kan fås fra direktoratet, J. nr. 30-00-00/80.

Klaus Færgemann

Kursus * *

Datalæreforeningens gymnasiefraction indbyder hermed til kursus fra tirsdag den 18. november kl. 17.00 til fredag den 21. november kl. 16.30. Bemærk, at tidspunktet er ændret i forhold til det tidspunkt, der er angivet i meddelelsen om kursus i Datalære nr. 4, september 1980. Kursets indhold fremgår af nedenstående program. Vi har søgt at tilrettelægge et kursus, som behandler generelle spørgsmål vedrørende edb, skole og samfund. En datamaskines opbygning og virkemåde, programmeringsteknik og lignende vil altså *ikke*

blive behandlet på dette kursus, og det kan i denne forbindelse bemærkes at edb-firmaer *ikke* er inviteret til at udstille. Programmet torsdag aften og fredag er præget af de vilkår, der nu gælder for datalære i gymnasiet og HF efter at direktoratet har besluttet, at datalære skal afvikles som selvstændigt fag. Undervisningsinspektør Ib Fischer Hansen, som er underskriver af skrivelserne om datalære, vil være til stede på kurset torsdag aften, hvor der er lejlighed til debat om datalære i gymnasiet og HF.

DATALÆRE – efteruddannelseskursus

EDB, skole og samfund

* Tid : Tirsdag d. 18. november kl. 17.00 til fredag d. 21. november 1980 kl. 16.30.

* Sted : Fuglsøcentret, 8420 Knebel.

PROGRAM :

- * Tirsdag
18.00-19.00 Indledning til kursus.
19.00-21.00 Bent Barfod og hans debatfilm om EDB: Midt i en overgang.
- * Onsdag
09.00-11.00 Henning Tjørnehøj, Fagbevægelsens forskningsråd.
13.00-18.00 Virksomhedsbesøg, automatisering og processtyring.
19.00-21.30 Jern- og metalindustriens sammenslutning - diskussionsoplæg.
- * Torsdag
09.00-11.30 Professor H. B. Hansen. RUC: EDB-teknik. Status og fremtidsperspektiver.
Diskussion.
13.00-18.00 Erik Måløe, HHÅ: Teknologivurdering - diskussionsoplæg.
19.00-21.30 Debataften, Datalære-undervisning i gymnasiet og HF. W. Kjellberg Christensen
Ib Fischer-Hansen.
- * Fredag
09.00-11.30 Datalære i tværfaglig sammenhæng. Gruppearbejde.
13.00-16.00 Pædagogisk diskussion og evaluering. Nedsættelse af arbejdsgrupper.

Deltagerantal : 20

Kursusudgifter : 570,- på dobbeltværelse
720,- på enkeltværelse

Fri vikar, rejseudgifter og diæter efter de sædvanlige regler

- * Tilmelding senest den 3. november 1980 til
Klavs Færgemann
Jernstøbervænget 15 - 4600 Køge
Tlf.: 03-66 07 47 eller 03-65 08 80.

For deltagere, der måtte ønske det kan der arrangeres transfer og overnatning fredag til lørdag i forbindelse med foreningens generalforsamling.

Data lære

I "Danske Kommuner" nr. 16/1980 var der et interview med undervisningsminister Dorte Bennedsen omkring forsøgsundervisningen i folkeskolen.

Blandt forskellige spørgsmål og svar fandtes følgende:

Pladsmangel

DK: - *Et af de steder, der har været ydret ønske om at få ændret folkeskoleloven, er antallet af valgfag, hvor man bl. a. har ønsket mulighed for at tage edb med. Er det en ændring, man kunne tænke sig at foretage efter de mange forsøg, der har været hermed?*

DB: - Jeg har fået mange henvendelser om datalære, der alle har talt for, at vi burde tage dette fag med i valgfagene. Jeg skal ikke lægge skjul på, at min sjæl er meget tvedelt på dette spørgsmål. Vi har en folkeskole med meget stor

pladsmangel, hvad fag angår. Vi har allerede lavet nogle ændringer, der er mere eller mindre lykkelige, men i alt fald klart har vist tendensen til mindre plads til de kreative og alle andre fag end de bogligt intelligensmæssige.

Argumentationen, jeg har hørt for at tage datalære ind, har gået på den teknologiske udvikling i samfundet og vigtigheden af at få et vist kendskab og et afslappet forhold til edb - en argumentation som jeg kan være meget enig i. Det er klart, at vores folkeskole også skal sætte eleverne i alt fald til en vis grad i stand til at klare sig i samfundet sidenhen, men lukker vi op for datalære, kan der være vældig mange andre erhvervsprægede ting, hvor man med rette kan sige, at det bør også findes i skolen. Vi må i den kommende tid meget nøje overveje de modstridende hensyn i dette spørgsmål inden der tages beslutning.

Til dette har vi fra Torsten Alf Jensen, SEL, Ollerup modtaget nedenstående indlæg.

Mystisk afmystificering og undervisningsministerens opfattelse af datalære

Man hører ofte, at det er et væsentligt formål med datalæreundervisningen at "afmystificere" datamaskinen og den elektroniske databehandling, men hvad er det egentlig, der skjuler sig under udtrykket afmystificering? Er det et nyt ord for indoktrinering? Er der overhovedet noget, som af de unge opfattes som mystisk? Hvad er det for en mystik, man vil til livs?

Man kunne forestille sig, at de unge opfattede datamaskinen som et overnaturligt væsen, og at det var undervisningens opgave at afsløre datamaskinen som et menneskeskabt produkt, der pænt og ordentligt følger de naturvidenskabelige love. En sådan undervisning ville være helt overflødig, for de unge opfatter allerede isenkrammet med stor nøgternhed og uimponerethed. Lærere der møder en datamaskine første gang f.eks. på et kursus, nærmer sig måske med en vis ærefrygt. De vil gerne have alt forklaret, før de trykker på den første tast. Eller måske har de besluttet en gang for alle at undgå at røre ved djævelskabet. Unge derimod trykker straks og ser hvad der sker. De er aldeles overbeviste om, at udstyret er idiot-sikret. Hvis noget bliver ødelagt ved at de trykker på tilgængelige taster, er det i hvert fald ikke deres skyld.

Det er altså ikke nødvendigt at afmystificere i den her nævnte forstand. Det er heller ikke rimeligt at fokusere så kraftigt på holdningen til selve udstyret. Man risikerer at mindske interessen for det, som er det centrale, nemlig holdningen til de strukturændringer i erhvervsliv og samfund, som en forøget anvendelse af udstyret medfører. Der er almindelig enighed om, at vi på grund af edb-teknikken står overfor en omvæltning i samfundet. En omvæltning som ofte sammenlignes med industrialiseringen i forrige århundrede. Det er klart det offentlige uddannelsessystems opgave at forberede de unge på disse strukturændringer, så de selv kan tage aktiv del i udformningen af den nye struktur. Når man lægger vægt på at "afmystificere" kunne man frygte, at man kom til at bagatellisere de meget væsentlige og alvorlige strukturændringer ved at sige: Se, det hele er kun svagstrøm og silicium. Dette svarer til, at man for hundrede år siden havde sagt: Industrialisering, dekvificering af arbejdskraften, elendige boligforhold i arbejderkvarterer, osv., det er jo ikke andet end et stålstempel, der trykkes frem og tilbage af ganske almindelig vanddamp.

I stedet for at afmystificere i den her nævnte forstand, burde man lægge vægt på at give de unge en holdning til de meget komplekse struktur-mæssige forhold, som vil være afgørende for det erhvervsliv og det samfund de skal fungere i.

Man kunne også tænke sig, at man ved mystik tænkte på den mærkelige og kunstige måde, hvorpå man ofte kommunikerer, når man anvender edb. I nogle systemer skal man skrive 1, når man

mener ja, eller 999 når man mener slut, eller man må acceptere at regnskabet er stillet op på den mest besynderlige måde med den første dato nederst osv. Hvis det er denne form for mystik man tænker på, er det næppe undervisningssektorens opgave at afmystificere. I stedet for at lære de unge at acceptere denne særegne kommunikation, burde man lære dem, at det er dem selv, der som brugere skal stille krav til systemerne og bestemme, hvordan regnskaberne skal stilles op og hvordan man staver til slut.

Hvis man prøver at få eleverne til at acceptere den her nævnte form for mystik, er man med til at understøtte en lille professionel gruppes alt for store indflydelse på udviklingen, og det er stik imod de mål, som er formuleret i offentlige uddannelsessystemer.

Ovenstående blev skrevet for et par år siden, efter at jeg igen ved et møde havde undret mig over den mystiske afmystificering. Det blev derefter sat ind i et ringbind med henblik på eventuel senere anvendelse. Lejligheden til denne anvendelse kom nu i en artikel i "Danske Kommuner", august 80, hvor undervisningsminister Dorthe Bennedsen i et interview siger: "Argumenterne jeg har hørt for at tage datalære ind, er gået på den teknologiske udvikling i samfundet og vigtigheden af at få et vist kendskab og et afslappet forhold til edb".

Nu har bagatelliseringen af det, som efter min fattelse er det væsentlige, altså nået ministerplan. I folkeskolens formålsparagraf stk. 3 står: Folkeskolen forbereder eleverne til medleven og medbestemmelse i et demokratisk samfund og til medansvar for løsning af fælles opgaver".

Dette kan ikke betyde andet, end at når man har viden om, at den teknologiske udvikling vil

kunne ændre vort samfund radikalt, så må man give eleverne baggrund for at påvirke denne ændring, og det indebærer, at eleverne må have en betydelig forståelse, som det bl.a. er folkeskolens opgave at bibringe eleverne.

Ministeren udtaler, at der er en tendens til, at de kreative fag og alle andre fag end de bogligt intelligensmæssige får mindre plads i folkeskolen. Jeg har aldrig helt forstået denne opdeling i kreative og bogligt intelligensmæssige fag. Er det kreativt eller bogligt intelligensmæssigt at give skriftligt udtryk for sine meninger og idéer? Uanset hvilken opdeling man vælger, er det beklageligt at skulle reducere fag, som anses for væsentlige. Man kommer imidlertid ikke uden om at prioritere, og det er her helt afgørende, at datalære indgår i vurderingen, ikke som en mindre betydningsfuld "erhvervspræget ting", men som det, det er, nemlig en behandling af konsekvenserne for samfundet af en særdeles hurtig teknologisk udvikling, som vil komme til at berøre de fleste samfundsfunctioener og ethvert job.

SIDSTE!

Statsminister Anker Jørgensen nævnte i sin åbningstale i Folketinget, at undervisningsministeren overvejede at lade Datalære indgå som valgfag i 8. og 9. klasse.

Lad os nu håbe, at det ikke bliver ved overvejelserne, og lad os ligeledes håbe, at de andre partier i Folketinget også kan indse, at det nu er tiden, at eleverne i folkeskolens ældste klasse får mulighed for at kunne vælge Datalære.

● ● ●

Aalborg kommunale skolevæsen har i de sidste år drevet datalæreundervisning og anvendt edb i undervisningen i et samarbejde med ASDA, Aalborg Seminariums Data Afdeling, der har leveret edb-kraften.

Nu er skolevæsenet imidlertid blevet selvforsynende idet man har oprettet DAKS, Data-centret ved Aalborg Kommunale Skolevæsen. DAKS er udstyret med en RC7000 på 128 Kb lager med 2 pladelagre og diverse perifert udstyr. Indtil videre er der 5 opkaldslinier foruden lokale terminaler. Yderligere råder DAKS over to RC702 microdatamater, som specielt skal anvendes til undervisningsformidling i forbindelse med specialundervisning.



Gammel Nyhed

Foreningen af matematiklærere ved gymnasieskoler og seminarier i Danmark afholdt møde den 18. 10. 1961 med et foredrag om regnemaskiner på programmet.

Foredraget blev holdt af Chr. Gram. Bagefter var der forevisning af Regnecentralen.

➔ **OBS! OBS!**

Stof til næste nummer af bladet skal være redaktionen i hænde senest mandag, den 12. januar 1981.

COMAL 80 - hvorfor og hvordan

af Børge Christensen

(Fortsat fra sidste nr.)

Rekursive procedurer

I slutningen af juli modtog jeg omsider den første kørende version af COMAL 80, og de program-eksempler, som ledsager artiklen, er skrevet og kørt på denne version. Jeg vil starte med et eksempel, som både demonstrerer nogle af de helt simple og nogle af de meget "dyre" udvidelser i forhold til de hidtil anvendte COMAL-versioner. Der er samtidig tale om et program, som læserne kender - eller let kan komme til at kende - teori og baggrund for. Der er nemlig tale om en *rekursiv* udgave af *quicksort*, og skulle man have glemte teorien for denne berømte algoritme, kan man passende grave DATALÆRE, nummer 2, 1979, frem af dyngen. På side 21 i dette udmærkede skrift begynder en af publikationens stjerneartikler under overskriften QUICKSORT (i det følgende henvises til denne artikel med: QS). Som man vil erindre eller kunne forvise sig om beror quicksort på følgende grundidé: En vektor, som skal sorteres, opdeles i to delvektorer på en sådan måde, at den ene del med sikkerhed ikke indeholder komponenter, der kommer før komponenterne i den anden del, og derpå udføres quicksort på de to delvektorer hver for sig. Processen er altså født rekursiv, og som gennemgangen i den ovenfor nævnte artikel viser, kræver det en del raffineret bogholderi af få processen omskrevet, så den kan udføres iterativt, altså ved hjælp af løkker. COMAL 80 udgaven af quicksort er vist i fig. 1. Opdelingen af vektoren foregår efter den samme algoritme, som er beskrevet i artiklen fra 1979 (QS, side 22), og i det nye program udføres den af det programafsnit, der indbefatter linjerne 100-220 (læseren opfordres *kraftigt* til at sammenligne algoritmen og programafnittet). Idet vi angiver hele opdelingsprocessen med ordene:

udfør opdeling

får vi følgende rekursive beskrivelse af quicksort:

proc quicksort

udfør opdeling

hvis der er en venstre delrække så

udfør quicksort på denne delrække

hvis der er en højre delrække så

udfør quicksort på denne delrække

endproc quicksort

En sammenligning af denne algoritme med den tidligere givne (QS, side 29) viser straks, hvor forbløffende simpel den nye er. Hvordan kan det

```

0010 /** PROGRAM: QUICKSORT **/
0020 /** SKREVET I COMAL-80, COMET VERSION **/
0030 /** EFTER M. WIRTH: ALGORITHMS-DATASTRUCTURES ECT. **/
0040 /** AF BØRGE R. CHRISTENSEN **/
0050 /** DATO: TØNDER, DEN 29. JULI, 1980 **/
0060 //
0070 PROC QUICKSORT(V, H)
0080 EXEC SNAPSHOT(V, H, A, N)
0090 I:=V I:=H
0100 X:=A((V+H) DIV 2) //SET PIVOT//
0110 REPEAT //OPDEL VEKTOR//
0120 WHILE A(I) < X DO
0130 I:=I+1 //RYK PILEN TIL HØJRE//
0140 ENDMHILE
0150 WHILE X < A(J) DO
0160 J:=J-1 //RYK PILEN TIL VENSTRE//
0170 ENDMHILE
0180 IF I <= J THEN
0190 W:=A(I); A(I):=A(J); A(J):=W //BYT A(I) OG A(J)//
0200 I:=I+1; J:=J-1 //OG SÆT PILE TIL NÆSTE POSITION//
0210 ENDIF
0220 UNTIL J < I
0230 IF V < J THEN EXEC QUICKSORT(V, J) //SORTER VENSTRE-DEL RÆKKE//
0240 IF I < H THEN EXEC QUICKSORT(I, H) //SORTER HØJRE-DEL RÆKKE//
0250 ENDPROC QUICKSORT
0260 //
0270 RANDOM
0280 INPUT "HVOR MANGE KOMPONENTER? ": N
0290 DIM A(N) //VEKTOR TIL SORTERING//
0300 TAB:=" "
0310 FOR I:=1 TO N DO
0320 A(I):=RND(1, 2*N)
0330 NEXT I
0340 FOR I:=1 TO N DO
0350 PRINT A(I),
0360 NEXT I
0370 PRINT
0380 SELECT LP:
0390 EXEC QUICKSORT(1, N)
0400 PRINT
0410 SELECT DS:
0420 FOR I:=1 TO N DO
0430 PRINT A(I),
0440 NEXT I
0450 END
0460 //
0470 PROC SNAPSHOT(VENST, HØJ, REF VEKTOR(), N) CLOSED
0480 VENST:=1; HØJ:=N
0490 PRINT " "
0500 FOR I:=1 TO N DO
0510 PRINT VEKTOR(I),
0520 NEXT I
0530 PRINT
0540 PRINT TAB((VENST-1)*4); " " ; TAB((HØJ-1)*4); " "
0550 PRINT
0560 ENDPROC SNAPSHOT

```

Fig. 1.

Eksempel på udskrift

```

16 18 11 8 17 15 17 10 9 11
16 11 11 8 9 15 10 17 17 18
8 11 11 16 9 15 10 17 17 18
8 11 11 10 9 15 16 17 17 18
8 9 10 11 11 15 16 17 17 18
8 9 10 11 11 15 16 17 17 18
8 9 10 11 11 15 16 17 17 18
8 9 10 11 11 15 16 17 17 18
8 9 10 11 11 15 16 17 17 18
9 10 11 11 15 16 17 17 18
8 9 10 11 11 15 16 17 17 18
8 9 10 11 11 15 16 17 17 18
8 9 10 11 11 15 16 17 17 18

```

være? De processer, der skal udføres, må da i grundens være de samme. Svaret er, at selvfølgelig udføres de samme processer, som beskrevet tidligere, men en væsentlig del af styringen af disse er nu overladt til COMAL-fortolkeren. Hele hemmeligheden ligger i, at *proceduren kan kaldes rekursivt*, eller sagt på en anden måde: proceduren kan "kalde sig selv". Sætningen:

*hvis der er en venstre delrække så
udfør quicksort på denne delrække*

er i programmet implementeret således (linje 230):

```
IF V < J THEN EXEC QUICKSORT(V, J)
```

At V er mindre end J, betyder netop at der er en delrække til venstre for pivoten, som skal sorteres. Denne delrække er angivet ved de to *parametre* V og J, som står i parentes efter procedurekaldet. Parametrene V og J kaldes i denne sammenhæng for de *aktuelle* parametre, og deres værdier angiver indeks for hhv. første og sidste komponent i delrækken. Værdien af disse to parametre overtages af de to *formelle* parametre V og H, som står anført i procedurehovedet (linje 70):

```
PROC QUICKSORT(V, H)
```

dvs. V og H har efter kaldet de samme værdier som V og J. Men pas på! Det V, der står i procedurehovedet, og det V, der står som aktuel parameter i kaldet, bliver af systemet opfattet som to *forskellige variable*, og det er netop en af hemmelighederne ved det hele. Hver gang, proceduren kaldes, opretter systemet en *ny udgave* af de to formelle parametre V og H, og det er denne række af udgaver af V og H, i hvilket de enkelte elementer kan have helt forskellige værdier, der erstatter den *stak*, som vi selv måtte opretholde i den iterative version af quicksort (se QS, side 23 - 28). Det er naturligvis ikke nogen helt simpel sag at få fortolkeren til at overtage bogholderiet, men det kan brugeren være ligeglad med. Den dertil svarende hovedpine er overstået én gang for alle med skrivningen af fortolkeren.

I samme nummer af DATALÆRE, som jeg har henvist til i det foregående, findes i øvrigt en udførlig gennemgang af begrebet: rekursiv procedure. Artiklen hedder "Tårnet i Hanoi" og er skrevet af H. B. Hansen. Den begynder på side 7 i det omtalte nummer.

Brug af parametre

Parametre kan naturligvis også bruges, uden at der behøver at være tale om rekursion, og i langt de fleste anvendelser er der tale om simple til-

fælde, hvor en procedure kaldes fra hovedprogrammet eller *en anden* procedure. Når jeg alligevel har valgt at bruge en rekursiv procedure som eksempel, hænger det sammen med, at dels er det i sådanne tilfælde, et system skal stå sin prøve, og dels, at jeg tidligere har gennemgået algoritmen i datalæreblandet. Jeg forstår følgelig selv dette eksempel og kan på bekvemteste vis henvise læserne til et arbejde, jeg har udført.

Programlisten er ledsaget af en udskrift, som er fremkommet ved kørsel af programmet. Denne udskrift administreres af proceduren SNAPSHOT (linje 470 - 580). I forhold til procedure QUICKSORT er der altså tale om et "assisterende program" (nyeste eufemisme for underprogram), og jeg vil ikke gennemgå den i sin helhed. Den indeholder imidlertid et par detaljer, som er karakteristiske for COMAL 80 og væsentlige for forståelsen af de idéer, der ligger bag denne nye udvidelse af COMAL. Hovedet af SNAPSHOT findes i linje 470 og ser sådan ud:

```
PROC SNAPSHOT(VENST, HØJ, REF VEKTOR (), N) CLOSED
```

og kaldet til SNAPSHOT findes i linje 80 og ser sådan ud:

```
EXEC SNAPSHOT(V, H, A, N)
```

De to formelle parametre VENST og HØJ får ved kaldet tildelt de samme værdier, som de aktuelle parametre V og H har. Noget tilsvarende gælder den formelle parameter N, der får samme værdi som den aktuelle parameter N, men der er igen grund til at understrege, at de to N'er af systemet opfattes og behandles som to *forskellige variable*. På listen over formelle parametre finder man også denne: REF VEKTOR(). Ordet "REF" er en forkortelse for "REFERENCE", der som bekendt betyder "henvisning". I forhold til tidligere COMAL-versioner er der tale om et helt nyt nøgleord. Når dette ord er anført foran en formel parameter, betyder det, at den pågældende parameter *henviser til det samme datafelt*, som den aktuelle parameter. Lad os se lidt nærmere på den her givne reference-parameter VEKTOR(). Den tomme parentes efter navnet betyder, at parameteren henviser til *én-dimensional vektor*, og ser vi efter i kaldet, kan vi konstatere, at den tilsvarende aktuelle parameter netop er navnet på en sådan én-dimensional vektor (A). Navnet VEKTOR bliver derved blot et *nyt navn for A*, sålænge der arbejdes *indenfor* rammerne af SNAPSHOT. Man kan sige, at VEKTOR er et "øgenavn" for A, som det er tilladt SNAPSHOT at bruge. Hver gang SNAPSHOT foretager sig noget med VEKTOR, er det i virkeligheden A, det "går ud over". - En dreng, der hedder Ole,

bliver kaldt Wolle af sin bror, så hver gang, Wolle er uvenner med sin bror, er det Ole, det går ud over.

I visse kredse, hvor man ynder at gøre sig ubegribelig på fremmede tungemål, siger man, at vektor bliver "called by reference", og om den anden type parametre, at de bliver "called by value". De tidligere nævnte parametre i COMAL III var alle henvisningsparametre. Jeg vil ikke på dette sted give flere og mere dybtgående eksempler på brugen af henvisningsparametre, men overlade til læseren selv at gøre sine erfaringer med dette redskab. For forfatteren er der imidlertid ikke nogen som helst tvivl om, at dersom der skulle vælges mellem værdi-parametre og henvisningsparametre, ville valget ubetinget falde på de sidste.

Lukket procedure

Procedurehovedet afsluttes med ordet "CLOSED". Tilstedeværelsen af dette nøgleord bevirker, som ordet selv siger, at proceduren er lukket. Det vil sige, at alle variable, der optræder i procedurens krop, er lokale for proceduren. Det betyder fx., at tælleren I, som optræder i linje 500 - 520, intet har at gøre med den tæller I, der optræder som pil i QUICKSORT. De to I'er opfattes af systemet som to forskellige variable. Lukkede procedurer er ganske vist ikke med i COMAL-80-kernen, men da de findes i både COMET-COMAL 80, RC-COMAL 80 og ID-COMAL 80, har jeg fundet det betimeligt at nævne begrebet her. I undervisningen kan de blive af meget stor betydning, idet man nu for første gang kan skrive en række biblioteksprocedurer som lukkede procedurer. Indholdet af disse "pakker" behøver ikke at være kendt i detaljer af brugeren, der blot skal vide, hvilke parametre, der kræves ved kaldet af dem. Vedkommende behøver ikke at bekymre sig om, hvad de variable i den lukkede procedure hedder, idet disse ikke på nogen måde kan komme til at påvirke de variable, som anvendes i det program, fra hvilket proceduren kaldes. Vi kan fx. tænke os, at man vil lukke proceduren QUICKSORT. Man skal da blot udstyre den med dette hoved:

```
PROC QUICKSORT(V, H, REF A()) CLOSED
```

Det er klart, at man er nødt til at anbringe henvisningsparameteren A() på parameterlisten, idet den vektor A, der optræder i proceduren, nu er en rent lokal vektor, der kun kan knyttes til hovedprogrammets A ved hjælp af henvisningen. Man behøver ikke længere forklare, hvad QUICKSORT er for noget, men kan nøjes med at give besked om, at dersom en tal-vektor skal sorteres, kan man blot føje en kopi af QUICKSORT - som

naturligvis står på programbiblioteket - til sit program og anvende et kald af formen:

```
EXEC QUICKSORT(FØRST, SIDST, VEKTOR)
```

For første gang er det virkelig muligt at bygge et helt COMAL-program op af "pakker", og begrebet "biblioteksprogrammer" får en helt anden og langt videre betydning, end det hidtil har haft i denne sammenhæng. Som man vil kunne se, er vi med begrebet lukket procedure meget nær ved at kunne benytte os af rigtige *externe procedurer*, og COMAL-gruppen har gjort sig overvejelser i den retning, men en række tekniske problemer er endnu ikke tænkt helt igennem.

Andre faciliteter

I øvrigt kan man i programmet se eksempler på følgende faciliteter i COMAL 80:

Kommentarer kan anbringes efter alle slags sætninger, og indledes med symbolet "//". Ordet REM kan stadig benyttes umiddelbart efter linjenumre, men vil af systemet blive ændret til "//", når programmet listes.

Der findes nu et rigtigt *tildelingstegn* i COMAL, nemlig det fra Algol og Pascal kendte ":=". Man kan stadig skrive fx. "A=5" eller "LET A=5", men systemet vil selv ændre disse to til "A:=5". Denne detalje kan måske synes betydningsløs, men det har vist sig, at brugen af det samme symbol for tildelingstegnet og lighedstegnet - som intet har med hinanden at gøre - ofte har ført til begrebsforvirring, især hos begyndere. Og hvad betyder i grunden følgende:

```
FUNDET=A=X(I)
```

En COMAL 80 fortolker ændrer selv dette til:

```
FUNDET:=A=X(I)
```

Der er endnu en grund til at bruge et særskilt tildelingstegn. I såvel COMET-COMAL 80 som RC-COMAL 80 kan man skrive fx.:

```
SALGSPRIS:+MOMS
```

og denne sætning har samme virkning som:

```
SALGSPRIS:=SALGSPRIS+MOMS
```

På tilsvarende måde kan ":-" bruges. Da "+" bruges ved *sammenkædning* af tekster (strenger), kan man også skrive fx.:

```
TEKST$ :+LINJES
```

i betydningen:

TEKST\$:=TEKST\$ + LINJE\$

I programmets linje 480 har jeg benyttet "+". Det skal dog bemærkes, at denne facilitet ikke er kommet med i COMAL 80 kernen. Man kan altså ikke forvente at finde den i alle COMAL 80 versioner. Når jeg alligevel har nævnt muligheden,

er det fordi det har indgået i de overvejelser, der er gået forud for valget af et egentligt tildelings-tegn.

I næste artikel vil jeg fortælle om bl. a. de nyheder der findes på områderne: vektorer, matricer og tekster.

Børge R. Christensen

Almindelige oplysninger om foreningen

Bestyrelsens sammensætning:

- Formand: ERLING SCHMIDT
Revlingebakken 40, II, 9000 Ålborg, tlf. (08) 18 53 66.
- Næstformand: WILLY KJELLBERG CHRISTENSEN
Strandpromenaden 32, 4900 Nakskov, tlf. (03) 92 30 34.
- Sekretær: FRITZ G. KNUDSEN
Kollerupvej 17, 8900 Randers, tlf. (06) 43 49 04.
- Kasserer: TORBEN HØIRUP
Karl Withsvej 2, 5000 Odense C, tlf. (09) 14 33 53.
- HUGO JØRGENSEN
Olivenvvej 11, Helsted, 8900 Randers, tlf. (06) 42 37 91.
- GERD BELHAGE
Slettebjergvej 7, 2750 Ballerup, tlf. (02) 97 10 46.
- TORSTEN ALF JENSEN
Langemarken 27, 5762 Vester Skerninge, tlf. (09) 24 22 35.

Henvendelser til foreningen:

Indmeldelser, adresseændringer o.l. til kassereren:

FORENINGEN FOR DATALÆRE OG ANVENDELSE AF EDB I
UNDERVISNINGEN
Rismarksvej 80, 5200 Odense V, tlf. (09) 16 86 50.

eller til privatadressen.

Årskontingent: 90 kr. incl. blad. Studerende 45 kr.

Øvrige henvendelser til formanden.

BLADET:

Ansvarshavende redaktør:

TEDDY LANG PETERSEN
Holstedvej 7, 5200 Odense, tlf. (09) 16 90 56.

Henvendelser vedr. annoncer/stof:

Til redaktøren.

BAKSPOR

Et Comal-system til løsning af opgaver af kombinatorisk natur

H. B. Hansen

Baksporing

Mange opgaver, navnlig dem der optræder inden for området kunstig intelligens, er kombinatoriske af natur. Herved forstås at man ikke kender "hurtige" algoritmer til at løse dem, men er henvist til at prøve sig frem. Et typisk eksempel på en sådan opgave er at lægge et puslespil. Spørgsmålet er om man kan finde generelle metoder til løsning af sådanne problemer ved anvendelse af en datamaskine. For en ret stor klasse af disse opgaver kan dette spørgsmål faktisk besvares bekræftende. De opgaver det drejer sig om er sådanne der på en eller anden måde kan abstraheres til at være specialtilfælde af følgende generelle opgave:

Man skal bestemme elementerne i en vektor x , der har n elementer. Når alle elementer opfylder visse krav er opgaven løst. I puslespileksemplet kan elementet $x(i)$ repræsentere lægning af den i 'te brik, og når alle n brikker er lagt er opgaven løst.

Den mest umiddelbare og naive måde at løse en sådan opgave på er at prøve alle muligheder systematisk. Det er den metode der hedder opregningsmetoden i ref. 1. Opregningsmetoden har imidlertid den uheldige egenskab at den i de fleste tilfælde fører til meget lange beregningstider. Hvis hvert element i x -vektoren f. eks. har m mulige værdier, så vil der være m^n muligheder at undersøge. Beregningstiden stiger altså eksponentielt med vektorens længde, og dette sætter snævre grænser for hvor store opgaver man kan løse i praksis. I puslespileksemplet ville beregningstiden stige med $n!$, idet første brik kan vælges på n måder, anden på $n-1$, og så fremdeles. Der er derfor behov for en mere effektiv fremgangsmåde end opregning.

En sådan fremgangsmåde er baksporing (engelsk: backtracking). Ideen i metoden kan lettest forklares når man befinder sig midt i løsningsprocessen. Lad os derfor antage at vi på en eller anden måde har fået valgt de første $k-1$ elementer i x :

$$(x(1), x(2), x(3), \dots, x(k-1))$$

Opgaven består nu i at vælge $x(k)$. I langt de fleste konkrete opgaver vil det gælde at antallet af måder hvorpå $x(k)$ kan vælges, er begrænset, og afhænger af de elementer der allerede er valgt.

Sammen med de allerede valgte $k-1$ elementer i x skal $x(k)$ udgøre en mulig sti til en løsning, dvs. det skal stadig være muligt at vælge resten af x 'erne, $x(k+1)$, $x(k+2)$, \dots , $x(n)$, på en måde der fører til en løsning. Vi skal i det følgende se eksempler på hvordan dette krav kan "dræbe" mange potentielle muligheder for $x(k)$. Måske begrænses mulighederne så kraftigt at der er nul måder at vælge $x(k)$ på. Hvis det er tilfældet må man konkludere at et af de forrige elementer, på trods af udvist forsigtighed, må være valgt forkert, og derfor "bakker man op" og reviderer valget af $x(k-1)$. Fordelen ved metoden er gemt i denne baksporing, idet alle de muligheder der ligger i forlængelse af det k 'te element, overhovedet aldrig prøves, og derfor ikke tager tid. Hvis der eksempelvis i teorien er m mulige valg for hvert element, og man bakker op ved nr. k , så overspringes m^{n-k-1} mulige vektorer helt.

8-dronningeproblemet

Et af de klassiske kombinatoriske problemer er 8-dronningeproblemet, der går ud på at anbringe 8 dronninger på et skakbræt, således at ingen af dem kan slå hinanden. Dette problem kan man formulere i baksporsterminologien på følgende måde:

De 8 dronninger kan nummereres fra 1 til 8, og da der ikke kan være to dronninger i samme række på skakbrættet, kan vi uden tab af generalitet forudsætte at dronning nr. k placeres i række nr. k . Alle løsninger kan derfor repræsenteres som en vektor med 8 elementer, $(x(1), x(2), x(3), \dots, x(8))$, hvor $x(k)$ er søjlenummer for dronning nr. k . Da to dronninger heller ikke kan stå i samme søjle må vi som den første betingelse forlange at alle x 'erne er forskellige. Den sidste betingelse vi må stille er at ingen af dronningerne står i samme diagonal (skrå linie) på skakbrættet.

Hvis skakbrættet repræsenteres som en todimensional tabel, DIM B(8,8), vil man meget hurtigt kunne overbevise sig om at to brikker på hhv. feltet B(i,j) og feltet B(k,l) står i samme diagonal såfremt:

$$i - j = k - l \quad \text{eller} \quad i + j = k + l$$

Disse to betingelser kan sammensættes til:

$$|j - l| = |i - k|$$

hvor $| \quad |$ står for numerisk værdi.

Hvis vi har givet de $k-1$ første elementer i x -vektoren, så vil en algoritme til bestemmelse af om en bestemt værdi kan bruges for $x(k)$ derfor kunne udformes omtrent således:

```
FOR værdi=1 TO 8
  hvis det gælder for alle i mellem 1 og k-1
  incl., at  $x(i)$  er forskellig fra værdi, og samtidig
   $|x(i)-værdi|$  er forskellig fra  $|i-k|$ ,
  så er værdi en ny mulighed for elementet
   $x(k)$ 
NEXT værdi
```

Comal-systemet BAKSPOR

At programmere en løsning til 8-dronningeproblemet ved baksparing er jo desværre ikke gjort ved blot at omsætte algoritmen i forrige afsnit til f. eks. korrekt Comal. Der hører en del programmering til, med administration af selve metoden, med regnskab med hvilke muligheder der er prøvet, hvornår der skal bakkes op, osv. Denne del af programmet er egentlig ganske uvedkommende for selve problemet, men er datalogisk set den vanskeligste del af det hele. Hertil kommer at den vil gå igen fra opgave til opgave, og det ville derfor være en stor fordel hvis den kunne programmeres én gang for alle.

```
0010 ENTER "BAKSPOR"
0020 INPUT "ANTAL DRONNINGER:";NTOT
0030 LET NKAN=NTOT
0040 EXEC BAKSTART
0050 REPEAT
0060   EXEC BAKSPOR
0070   IF LØSNING THEN MAT PRINT SVAR
0080 UNTIL FÆRDIG
0090 PROC KANDIDAT
0100   LET N=NTOT
0110   IF K=1 THEN LET N=(NTOT+1) DIV 2
0120   FOR MULIG=1 TO N
0130     LET I=1; OK=1
0140     WHILE I<K AND OK DO
0150       IF SVAR(I)=MULIG OR
          ABS(SVAR(I)-MULIG)=ABS(I-K) THEN
0160         LET OK=0
0170       ELSE
0180         LET I=I+1
0190       ENDIF
0200     ENDWHILE
0210     IF OK THEN EXEC GEM
0220   NEXT MULIG
0230 ENDPROC
0240 END
```

Fig. 1

Det er netop det BAKSPOR-systemet sørger for. Som et eksempel vil jeg nu gennemgå en løsning af dronningeproblemet ved brug af BAKSPOR, se fig. 1. I linie 10 hentes systemet fra baggrundslager, og i linie 20 indlæses problemets størrelse, dvs. længden af problemets svarvektor. Denne længde er en systemvariabel ved navn NTOT. Programmet løser altså dronningeproblemet med NTOT dronninger. Den variable NKAN i linie 30 er også en systemvariabel; den skal overfor systemet angive antallet af mulige kandidater pr. trin i løsningsprocessen. Her er NKAN=NTOT fordi hver ny dronning principielt kan

anbringes på NTOT felter. Disse to størrelser skal bruges til at dimensionere systemets tabeller, hvilket sker ved kald af systemproceduren BAKSTART i linie 40. Fra linie 50 til linie 80 er der en løkke der kalder systemproceduren BAKSPOR. Denne procedure er hjertet i systemet. Den opererer på en systemvektor ved navn SVAR med dimension NTOT; SVAR er den vektor vi hidtil har kaldt x -vektoren. Hver gang BAKSPOR har fundet en ny løsning, sættes systemvariablen LØSNING til 1, og når der ikke er flere løsninger bliver systemvariablen FÆRDIG lig med 1. Ved hjælp af løsningsløkken fra linie 50 til linie 80 finder man altså alle løsninger på NTOT-dronningeproblemet.

BAKSPOR tager sig kun af det rent administrative, og derfor må "spillereglerne" i problemet udformes af brugeren, i form af en procedure ved navn KANDIDAT, som BAKSPOR vil kalde på passende tidspunkter. KANDIDAT for dronningeproblemet er vist på fig. 1, linie 90 til 230. Kerner i proceduren, det der afspejler algoritmen i forrige afsnit, findes i linie 120 til linie 220. MULIG er en systemvariabel der bruges af systemproceduren GEM (se linie 210) til at gemme de mulige kandidater for SVAR(K), når SVAR(1) til SVAR(K-1) er givet. WHILE-løkken fra linie 140 til linie 200 undersøger om betingelsen for kandidatur er opfyldt, og signalerer dette i variablen OK, således, at de "gode" værdier af MULIG kan gemmes i linie 210. Bemærk at K også er en systemvariabel, og derfor ikke må bruges til andre formål af brugeren.

3	5	2	4	1
3	1	4	2	5
2	5	3	1	4
2	4	1	3	5
1	4	2	5	3
1	3	5	2	4

Fig. 2

Fig. 2 viser programmets løsning af 5-dronningeproblemet, dvs. NTOT=5. Hvis man tegner løsningerne ind på et 5*5 skakbrædt vil man opdage at der mangler alle de løsninger der er symmetriske med de fundne løsninger. Denne besparelse er opnået ved hjælp af de to programlinier 100 - 110 på fig. 1. De udtrykker at hvis det er første række (K=1), så skal man kun tage det halve antal muligheder med. Herved reduceres arbejdet - og dermed den samlede beregningstid - til det halve, men tiden i KANDIDAT bliver en anelse længere.

Løsning af cryptaritmer

I ref. 2 lanceres glosen "cryptaritmer" for de velkendte problemer af typen:

Fortsættes side 21.

METRIC PRÆSENTERER:

METRIC ABC80

80'ernes Mikrodatamat- system



METRIC ABC 80 - MIKRODATAMATEN til morgendagens undervisning!

- * Let at lære - nem i brug
- * Lynhurtigt og avanceret semikompilerende BASIC, det internationale sprog
- * 16K bruger-RAM (opt. 32K)
- * Kassettebåndstation til program- og datalagring
- * Skandinavisk kvalitetsprodukt
- * 16K ABC 80 BASIC i ROM, altid klar til brug, ingen besværlig indlæsning
- * 12" dataskærm med grafik
- * Professionelt tastatur
- * Har udgang for modem, kan også bruges som terminal
- * Pris: 8.990,- excl. moms

DATAUDSTYR FRA SC **METRIC** AIS

DATAAFDELINGEN, SKODSBORGVEJ 305, 2850 NÆRUM, TLF. (02) 80 42 00

DET VIGTIGSTE...



Lærebøger der passer til datamaten

- * Pædagogisk udformede lærebøger der hjælper eleven til hurtigt at forstå datamaten. På dansk - selvfølgelig
- * Også bøger for den, der vil vide mere om datamatens indretning, måledatamatsystemer og programdokumentation
- * Ingen uoverensstemmelser mellem datamat og lærebog
- * Bestil et sæt - læs - og bliv overbevist

Videointerface

- * Lader hele klassen se med på en datamat
- * Tilsluttes skolens videoanlæg eller et almindeligt TV
- * Pris: 650,- excl. moms

Priseksempler:

Grundsystem m. CPU, tastatur, dataskærm og kassette	8.990,-
System med grundsystem, dobbelt floppy disk (2 x 80K), ABC P80 printer og interface for denne	28.990,-
Klassesæt m. 6 x grundsystem og ABC P80 printer med omskifter for 6 x ABC 80	63.940,-

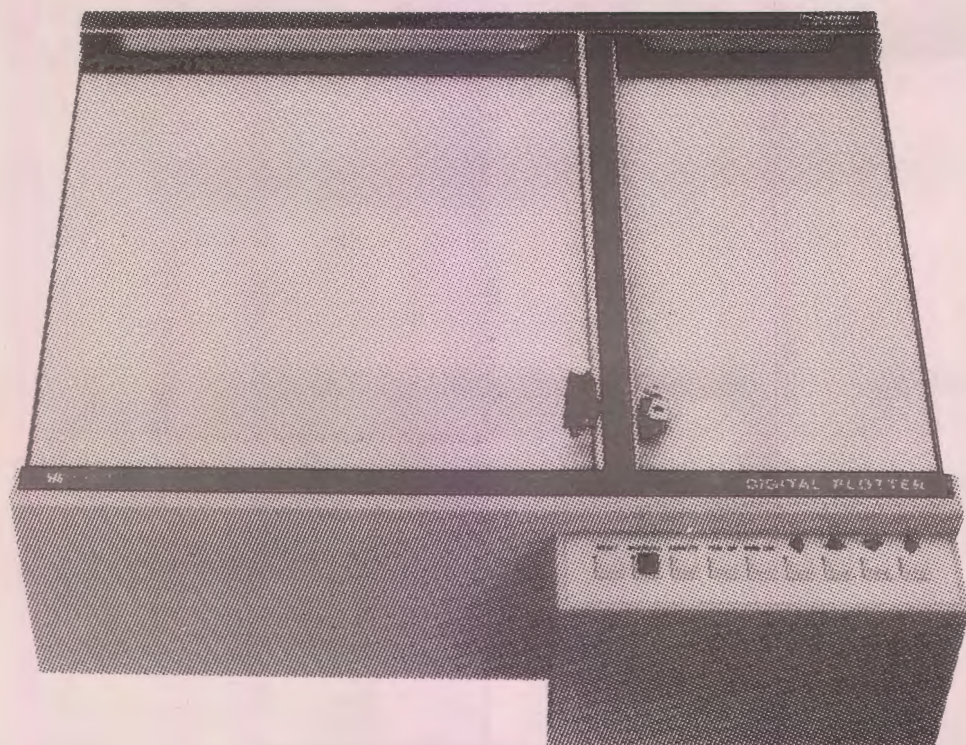
Alle priser forstås excl. moms

DATAUDSTYR FRA SC **METRIC** AIS

DATAAFDELINGEN, SKODSBORGVEJ 305, 2850 NÆRUM, TLF. (02) 80 42 00

INTELLIGENT PLOTTER!

A3 format. HOUSTON DMP-7, kun 16.300,-
excl. moms



- Plotning af vektorer
- Automatisk udskrift af tal, bogstaver og tegn med variabel størrelse og retning
- Hastighed: 50 mm/sek.
- Tilsluttes V-24/RS 232
- Indbygget selvtest
- Plotning i absolutte x-y koordinater
- Automatisk udskrift af 9 forskellige linietyper
- Opløsning: 0,1 mm
- Format: 254 x 380 mm
- Styres med simple ASCII koder
- Forlang special-prospekt

FRA DET STORE HOUSTON PROGRAM I ØVRIGT:

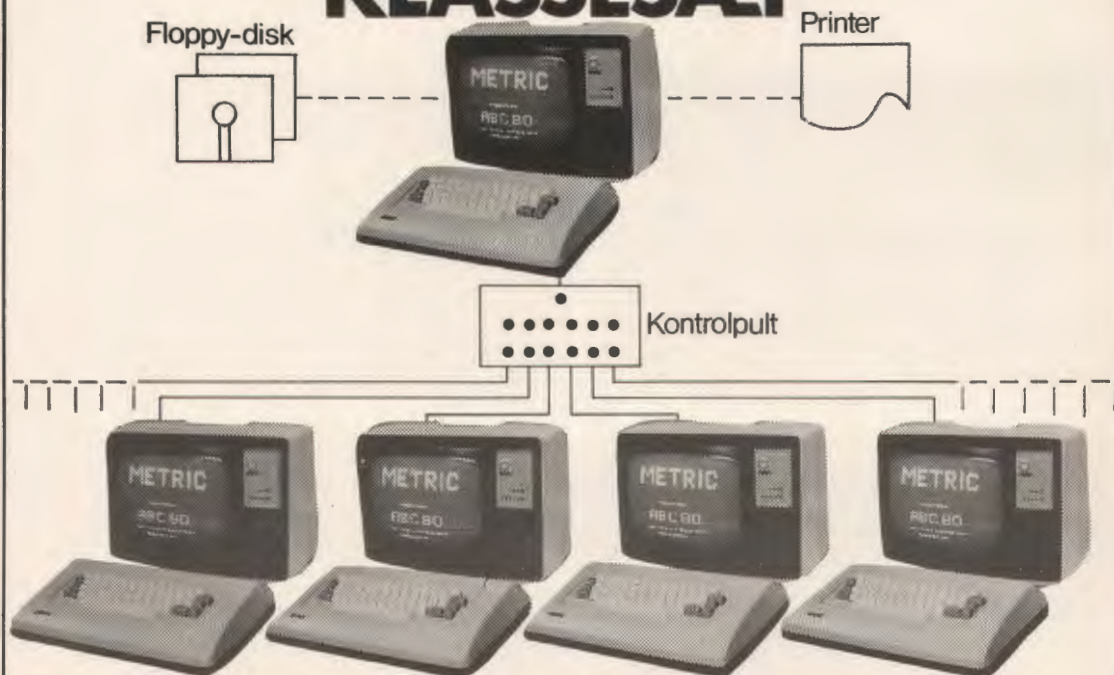
- DMP-2: A4 ikke intelligent plotter
Kr. 9.600,- excl. moms
- HI-PAD: 28x28 cm digitizer
Kr. 7.500,- excl. moms

DATAUDSTYR FRA SC METRIC AIS

DATAAFDELINGEN, SKODSBORGVEJ 305, 2850 NÆRUM, TLF. (02) 80 42 00

METRIC ABC80

KLASSESÆT



- Lærerdatabmat med kontrolpult
- Op til 24 elevdatamater
- Læreren kan sende programmer og opgaver til en eller flere elever samtidig

80'ERNES MIKRODATAMAT

DATAUDSTYR FRA SC **METRIC** AIS

DATAAFDELINGEN, SKODSBORGVEJ 305, 2850 NÆRUM, TLF. (02) 80 42 00

SEND
+ MORE
MONEY

Jeg vil nu vise hvordan sådanne problemer kan løses ved hjælp af BAKSPOR, ved at finde en løsning til ovennævnte opgave.

Det der efterlyses er værdierne af elementerne i en vektor, hvor hvert element betegner et af bogstaverne i problemet. Der er otte forskellige bogstaver, og det er ligegyldigt i hvilken rækkefølge vi finder dem, så lad os vælge rækkefølgen MOSENDRY. Vi har altså $NTOT=8$. Ethvert element kan i princippet antage enhver af de ti mulige værdier fra 0 til 9, så $NKAN=10$. Vi er nu i stand til at programmere hovedprogrammet for opgaven, se fig. 3. Det eneste "mærkelige" i dette program er linie 621 og 622, som jeg har indsat for at kunne føre regnskab med algoritmens effektivitet, herom mere senere.

```
0580 ENTER "BAKSPOR"
0590 DIM BOGST*(8)
0600 READ BOGST*
0610 DATA "MOSENDRY"
0620 LET NTOT=8: NKAN=10
0621 DIM ANTAL(NTOT)
0622 MAT ANTAL=ZER
0630 EXEC BAKSTART
0640 REPEAT
0650 EXEC BAKSPOR
0660 UNTIL LØSNING
0670 FOR I=1 TO 8
0680 PRINT BOGST*(I,I); "="; SVAR(I); "ANTAL="; ANTAL(I)
0690 NEXT I
0700 END
```

Fig. 3

Tilbage er proceduren KANDIDAT. Denne procedure kunne naturligvis programmeres ganske lige ud ad landevejen, sådan at alle muligheder der ikke allerede findes i SVAR, gemmes som nye muligheder. Dette ville imidlertid føre til lang beregningstid fordi en masse af de således gemte muligheder ville være forkerte. Og vi ved jo meget mere om sagen; regnestykket påtvinger nogle bånd mellem de ubekendte, som vi kan tage højde for, og derved eliminere en masse potentielle muligheder på forhånd. F. eks. må det jo gælde at $D+E=Y$, eller eventuelt $D+E=10+Y$ hvis der kommer mente ved additionen.

Nu er spørgsmålet hvor langt man skal gå i denne analyse. Går man helt til bunds, så er problemet løst, og der er ingen grund til at bruge datamaskine, men går man ikke langt nok bliver køretiden for stor. Proceduren KANDIDAT på fig. 4 er resultatet af 15 til 20 minutters analyse af de mest indlysende sammenhænge i problemet. Hvert WHEN i CASE-konstruktionen fra linie 20 til linie 440 tager sig af et bestemt bogstav, nemlig hhv. M, O, S, E, N, D, R, og Y. Som man ser forudsættes det at M er lig med 1, og at O er lig med nul, mens S kan være enten 8 eller 9. Dette kan man hurtigt finde ud af må være tilfældet

```
0010 PROC KANDIDAT
0015 LET ANTAL(K)=ANTAL(K)+1
0020 CASE K OF
0030 WHEN 1
0040 LET MULIG=1
0050 EXEC GEM
0060 WHEN 2
0070 LET MULIG=0
0080 EXEC GEM
0090 WHEN 3
0100 FOR MULIG=8 TO 9
0110 EXEC GEM
0120 NEXT MULIG
0130 WHEN 4
0140 FOR MULIG=2 TO 9
0150 EXEC NYKAN
0160 IF NYKAN THEN EXEC GEM
0170 NEXT MULIG
0180 WHEN 5
0190 LET MULIG=SVAR(4)+1
0200 EXEC TEST
0210 WHEN 6
0220 FOR Y=2 TO 9
0230 LET MULIG=Y-SVAR(4)
0240 EXEC TEST
0250 LET MULIG=MULIG+10
0260 EXEC TEST
0270 NEXT Y
0280 WHEN 7
0290 LET MULIG=SVAR(4)-SVAR(5)
0300 EXEC TEST
0310 LET MULIG=MULIG-1
0320 EXEC TEST
0330 LET MULIG=MULIG+11
0340 EXEC TEST
0350 LET MULIG=MULIG-1
0360 EXEC TEST
0370 WHEN 8
0380 LET MULIG=SVAR(6)+SVAR(4)
0390 EXEC REGN
0400 IF OK THEN EXEC TEST
0410 LET MULIG=MULIG-10
0420 EXEC REGN
0430 IF OK THEN EXEC TEST
0440 ENDCASE
0450 ENDPROC
0460 PROC TEST
0470 IF MULIG>1 AND MULIG<10 THEN
0480 EXEC NYKAN
0490 IF NYKAN THEN EXEC GEM
0500 ENDIF
0510 ENDPROC
0520 PROC REGN
0530 LET Y=SVAR(6)+10*(SVAR(5)+10*(SVAR(4)+10*SVAR(3)))
0540 LET Z=SVAR(4)+10*(SVAR(7)+10*(SVAR(2)+10*SVAR(1)))
0550 LET W=MULIG+10*(SVAR(4)+10*(SVAR(5)
+10*(SVAR(2)+10*SVAR(1))))
0560 LET OK=(W=Y+Z)
0570 ENDPROC
```

Fig. 4

ved at betragte opgaven. For bogstavet E (WHEN 4) forudsættes kun at det ikke kan være 0 eller 1, men herudover skal dets værdi jo være forskellig fra alle de hidtil valgte elementer i SVAR. Dette kan testes ved hjælp af systemproceduren NYKAN, der kaldes i linie 150 og som returnerer 0 hvis MULIG findes i SVAR i forvejen, og ellers 1. Denne test på om en værdi allerede findes i forvejen forekommer så hyppigt i forbindelse med baksportsproblemer, at det er praktisk at udarbejde en standardprocedure.

WHEN 5 vedrører bogstavet N. Fra opgaven ved vi at $E + O = N$, bortset fra eventuelle indgående og udgående menter. Da O er nul, og E og N skal være forskellige, så må der faktisk være en indgående mente, men der kan ikke være nogen udgående mente, for $E + 1$ kan højst blive 10, og N må ikke være nul. Dette forklarer linie 190, idet SVAR(4) jo er den øjeblikkelige værdi af E.

Proceduren TEST, der kaldes i linie 200, er indført af praktiske grunde; den kan ses i linie 460 til 510. Den undersøger om MULIG nu også er en reel mulighed, og gemmer den hvis det er tilfældet.

Bogstav D, When 6, behandles på grundlag af den øjeblikkelige værdi af E, samt de eksisterende muligheder for bogstavet Y. Bogstav R, WHEN 7, er relativt godt bestemt, idet såvel E som N findes i forvejen. Her er der derfor kun fire potentielle muligheder, når man tager menterne i betragtning. Endelig er der bogstavet Y, WHEN 8, hvor der kun er to muligheder, men her skulle hele regnestykket gerne gå op, hvilket testes i proceduren REGN, se linie 520 til 570.

Effektiviteten af BAKSPOR

Ved at bruge den direkte opregningsmetode ville man skulle igennem mange muligheder for at løse den foreliggende cryptaritme. Det nøjagtige antal kan man finde ved at bemærke at første bogstav kan vælges på 10 måder, næste på 9, næste igen på 8, og så fremdeles. Antallet af måder bliver derfor $10 \cdot 9 \cdot 8 \cdot 7 \cdot 6 \cdot 5 \cdot 4 \cdot 3 =$ ca. 1.8 millioner. Proceduren KANDIDAT begrænser dette antal betydeligt. Her undersøges alle kombinationer af de muligheder der er aktuelle for hver WHEN-sætning, dvs. $1 \cdot 1 \cdot 2 \cdot 8 \cdot 1 \cdot 16 \cdot 4 \cdot 2$, hvilket er 2048, men mange af disse forkastes allerede i selve KANDIDAT, fordi de findes i forvejen (i proceduren NYKAN). BAKSPOR vil yderligere reducere antallet, idet den vil opdage at visse af de tidligere valg umuliggør en senere bestemmelse af værdier, hvorfor den vil bakke op, og dermed overspringe nogle muligheder, som simpelthen aldrig bliver testet.

For at undersøge disse forhold noget nærmere indsatte jeg nogle tælleværker i programmet - et for hver komponent i SVAR-vektoren. På fig. 3, linie 621 og 622, defineres og nulstilles disse tælleværker, og på fig. 4, linie 15, tælles der op i det af dem der svarer til den øjeblikkelige K-værdi. Herved opsummeres antallet af gange KANDIDAT bliver kaldt, fordelt på de otte værdier af K. Resultatet af en kørsel ses på fig. 5; som man ser sker der en betydelig reduktion af de 2048 muligheder.

M=	1	ANTAL=	1
O=	0	ANTAL=	1
S=	9	ANTAL=	1
E=	3	ANTAL=	1
N=	6	ANTAL=	4
D=	7	ANTAL=	3
R=	8	ANTAL=	11
Y=	2	ANTAL=	6

Fig. 5

Da KANDIDAT jo gemmer mere end én MULIG for visse K-værdier, kørte jeg også med en version der talte antallet af kald af systemproceduren GEM. Resultatet var 32, altså en reduktion svarende til $2048/32 = 64$ gange.

Man kan altså konkludere at baksparing under visse omstændigheder kan føre til ganske betydelige reduktioner i det antal muligheder der prøves. Man kan godt selv hjælpe lidt med i denne reduktionsproces. I det foreliggende tilfælde ligger en del af hemmeligheden f. eks. i den rækkefølge hvori bogstaverne behandles. Dem med få muligheder er tildelt lave K-værdier, og det betyder at "forgreningerne" starter på et sent trin af løsningsprocessen, og derfor ikke når at blive multipliceret op.

Også den rækkefølge hvori kandidaterne gemmes, spiller en rolle. F. eks. fremgår det af fig. 5 at $S=9$ er fundet ved kun et enkelt kald af KANDIDAT, selv om MULIG har haft både værdien 8 og 9 for $K=3$ (se fig. 4, WHEN 3). Da $MULIG=9$ er gemt sidst, men øjensynligt prøvet først, ser man at de kandidater man tror mest på bør gemmes senere end dem man ikke har fidus til. Forklaringen er at BAKSPOR anvender en stak (se f. eks. ref. 3) til at gemme kandidaterne i før de flyttes over i SVAR. Af samme grund findes der i systemet to "hemmelige" variable ved navn STAK og S, som på grund af Comals lidt primitive natur ikke kan skjules helt for brugeren; man må derfor ikke bruge navnene STAK og S når man anvender BAKSPOR-systemet.

BAKSPOR-programmet

Comalprogrammet for BAKSPOR fremgår af fig. 6. Som man ser er der tale om et beskedent omfang i antal Comallinier, men programmeringsstilen er ret kompakt. Den nærmere analyse af detaljerne i systemets virkemåde overlades til den interesserede læser, idet jeg dog til slut vil påpege enkelte detaljer der kan have interesse for eventuelle brugere.

Systemproceduren BAKSPOR vil returnere til det program der har kaldt den hver gang K ændres (enten op eller ned). De systemdefinerede variable LØSNING og FÆRDIG vil da have relevante værdier, således at LØSNING er 1 hvis $K=NTOT$, mens FÆRDIG er 1 hvis $K=0$. Ved hver returnering til hovedprogrammet vil de første K elementer i SVAR indeholde en partiel løsning til den stillede opgave. I hovedprogrammet kan man derfor følge løsningsprocessens fremadskriden. En mulig anvendelse af dette kunne være at man løbende holder rede på brikkernes placering på et skakbræt, et puslespil eller lignende. En anden nærliggende anvendelse er at man vælger NTOT som en overgrænse for løsningsvektoren, således at der eksisterer løsninger med færre elementer - hvis antal måske er ubekendt. I dette tilfælde kan man i hovedprogrammet ignorere værdien af LØSNING, og selv teste om SVAR-vektoren indeholder en løsning efter hvert kald af BAKSPOR.

Standardsystemet BAKSPOR fritager brugeren

```

9000 PROC BAKSPOR
9010 LET NKAN=0; LØSNING=0
9020 WHILE NKAN=0 AND K>0 DO
9030 LET NKAN=STAK(S); S=S-1
9040 IF NKAN=0 THEN LET K=K-1
9050 ENDMHILE
9060 LET FERDIG=(K=0)
9070 IF NOT FERDIG THEN
9080 LET SVAR(K)=STAK(S); STAK(S)=NKAN-1
9090 LET LØSNING=(K=NTOT)
9100 IF NOT LØSNING THEN
9110 LET K=K+1; NKAN=0
9120 EXEC KANDIDAT
9130 LET S=S+1; STAK(S)=NKAN
9140 ENDF
9150 ENDF
9160 ENDPROC
9170 PROC GEH
9180 LET S=S+1; STAK(S)=MULIG; NKAN=NKAN+1
9190 ENDPROC
9200 PROC BAKSTART
9210 DIM SVAR(NTOT); STAK(NTOT*NKAN+1)
9220 LET K=1; S=0; LØSNING=0; FERDIG=0; NKAN=0
9230 EXEC KANDIDAT
9240 LET S=S+1; STAK(S)=NKAN
9250 ENDPROC
9260 PROC NYKAN
9270 LET NYKAN=1
9280 WHILE NYKAN>0 AND NYKAN<K DO
9290 IF SVAR(NYKAN)=MULIG THEN
9300 LET NYKAN=0
9310 ELSE
9320 LET NYKAN=NYKAN+1
9330 ENDF
9340 ENDMHILE
9350 LET NYKAN=(NYKAN=K)
9360 ENDPROC

```

Fig. 6

for at spekulere på alle de administrative detaljer ved løsning af en bestemt, ret vid, klasse af opgaver af kombinatorisk karakter. Derved udgør systemet en hjælp for rutinerede programmører, således at det bliver muligt at koncentrere opmærksomheden om det væsentlige i problem-løsningsprocessen, nemlig at beskrive kernen i et forelagt problem algoritmisk. Dette kan have betydning i en undervisningssituation, hvor ele-

verne kan klare større, og dermed mere realistiske, opgaver, hvis de fritages for at tænke på alle de datalogiske spidsfindigheder. Systemet er ikke et standardprogram eller underprogram i sædvanlig forstand, men udgør snarere en begrebsramme eller referenceramme, hvori man kan udtrykke sig når man vil løse baksporingsproblemer.

Idéen til denne måde at lave standardsystemer på skyldes Keld Helsgaun, der i en artikel andetsteds i bladet opstiller et system til løsning af differentialligninger. Inspirationen til at anvende denne idé på baksporing stammer fra ref. 5, hvori der beskrives et Fortran-system der ligner BAKSPOR.

Referancer

1. H. B. Hansen, Ole Caprani & Frank Jensen: Programmering og problemløsning - grundbog i datalære. Gyldendal 1978. Kapitel 5.
2. DATALÆRE, 4. årgang nr. 3, maj 1980, side 17.
3. H. B. Hansen: Tårnet i Hanoi, analyse af en rekursiv algoritme. DATALÆRE 3. årgang nr. 2, februar 1979.
4. Keld Helsgaun: SOFIE - et simpelt COMAL-system til numerisk løsning af differentialligninger. Roskilde Universitetscenter, Data-logiafdelingen, maj 1980.
5. Albert Nijenhuis & Herbert S. Wilf: Combinatorial Algorithms. Academic Press, New York 1975.

Husk! - Generalforsamling

* LØRDAG DEN 22. NOVEMBER 1980 KL. 11.30.
på Silkeborg Amtsgymnasium, Oslovej 10, Silkeborg.

Dagsorden i følge vedtægterne.

I tilknytning til generalforsamlingen arrangeres følgende:

* FREDAG D. 21. NOVEMBER
Kl. 19.00 Spisning med efterfølgende kollegialt samvær med mulighed for udveksling af erfaringer.
Kørsel på medbragte microer.

* LØRDAG D. 22. NOVEMBER
Kl. 9.00 - 10.00: Fraktionsmøder.
Kl. 10.00 - 11.30: Foredrag
Kl. 11.30 - : Generalforsamling.

Tilmelding til spisning fredag og/eller frokost lørdag er nødvendig.
Kan foretages hos: Jørgen F. Hansen tlf. (06) 81 24 47.

RC * INFO

RC-NYHEDSORGANET FOR RC-BRUGERE

PRODUKTCHEF

RC 700 er et af de hurtigst voksende produkter i Regnecentralens produktlinie. Og nok er undervisningssektoren et stort område for RC 700, men der er sandelig også bud efter den fra mange andre sider.

I takt med væksten bliver bemandingen selvfølgelig sat op, og nu har man for at få styring og samling på aktiviteterne i forbindelse med RC 700 udnævnt en produktchef.

Kaj Riss vil fremover varetage en række vigtige funktioner i forbindelse med videreudviklingen af RC 700, og han vil være den interne koordinator for så godt som alt vedrørende RC 700.

RC 702 I FINLAND

Der har været stor rift om de første eksemplarer af RC 702, der jo blev vist første gang for offentligheden på Fredericia-messen.

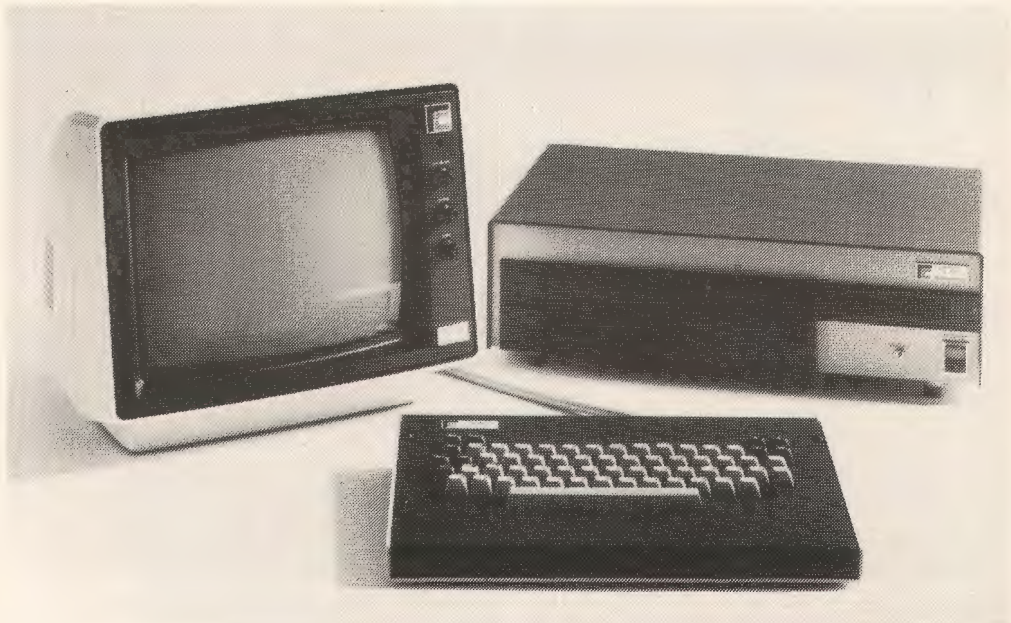
Den første store produktionsserie er på vej ud til køberne nu, men de allerførste "hånd-lavede" RC 702'ere har været sendt hid og did i ind- og udland.

Sidst har RC 702 været i Finland på en stor skoleudstilling, men det var desværre samtidig med, at den samme maskine skulle have været i Belgien. Skandinavien gik dog af med sejren, så belgierne fik "kun" lov at se et billede af RC 702, men fik dog lejlighed til at køre både COMAL og PASCAL på en RC 701.

FLERE AKTIER I RC 79

Det går den rigtige vej med RC 79, og fornylig har man udvidet aktiekapitalen fra 60 millioner kr. til 75 millioner kr.

Aktierne i Regnecentralen er i øvrigt stort set delt på tre grupper, der hver har ca. en trediedel af aktierne. De tre grupper er: det offentlige, telefonselskaberne og private firmaer.



RC 861 PRINTER

Printeren til RC 700 microdatamatsystemet stammer som monitoren fra Japan, og det er en særdeles fin lille ting, man har fundet frem.

Den kan skrive 100 tegn/sek. og med 80 tegn pr. linie ved normalt tegnsæt. Men man kan

også få den til at skrive med komprimeret skrift, hvorved der bliver plads til 132 tegn pr. linie. Endelig kan man også skrive med forstørrede bogstaver, hvoraf der kan være 40 på en linie.

Tegnsættet indeholder både store og små bogstaver og selvfølgelig også Æ, Ø og Å samt bogstaver med umlaut.

FØLGENDE SKRIFTTYPER FINDES PÅ RC 861 :

NORMAL SKRIFT : 80 tegn/linie

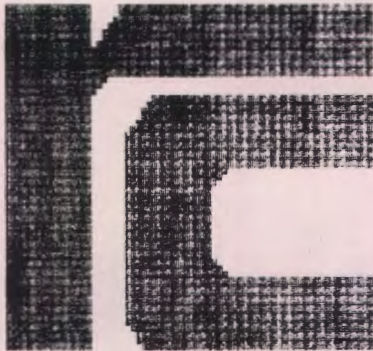
FORSTØRRET SKRIFT : 40 tegn/linie

KOMPRIMERET SKRIFT : 132 tegn/linie

TEGNSÆT (STORE/små BOGSTAVER) :

!"#\$%&'()*+,-./0123456789:;<=>?üABCDE

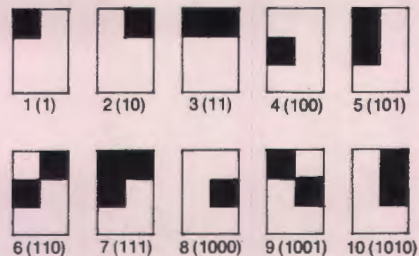
FGHIJKLMNOPQRSTUVWXYZÆØÅ+_äåbcdefghijklmno
pqrstuvwxyzæøåö



RC 700 MICRODATAMATSYSTEM

Men foruden at skrive kan RC 761 også tegne! Printeren kan nemlig sættes i grafisk mode, og her kan der så tegnes med et semi-grafisk tegnsæt, hvor tegnene når helt sammen. Det semigrafiske tegnsæt er opbygget på sædvanlig vis, d.v.s. der findes 63 tegn

opbygget efter nedenstående model (Faktisk findes der 64 semigrafiske tegn idet mellemrum skal regnes med)



1	2
4	8
16	32

OTHELLO

På indkøbsmessen for Stat, Amt og Kommune i Fredericia løb en "historisk" duel af stablen.

I nærheden af Regnecentralen udstilledes bl. a. ABC 80, og på et tidspunkt hvor der var lidt stille med gæster, blev der arrangeret en match i OTHELLO mellem RC 700 og ABC 80.

I OTHELLO spilles med brikker, som har forskellig farve på hver side, og spillerne skiftes til at lægge brikkerne ned på brædtet med deres farve opad. "Indkredses" der en række af modstanderens farve, vendes disse brikker og får derved ens egen farve.

I spillet i Fredericia førte ABC 80 til at begynde med, men RC 700 tog alvorligt fat tilsidst og vandt sikkert! Årsagen hertil er nok, at ABC 80'eren gik efter at vende flest brikker "her og nu", mens RC 700 var mere langsigtet i sin strategi og bl. a. gik efter kanterne af brædtet, hvilket giver store fordele i slutspillet.

OTHELLO programmet på RC 700 stammer fra Randers, hvor det er lavet af en meget kvik elev, der vist nok blev inspireret til at lave programmet netop efter at have set versionen på en ABC 80.

RC 702 I JAPAN

RC 700 microdatamaterne kommer langt omkring, og nu er en RC 702 havnet i Japan.

Forklaringen er, at den nye professionelle datamonitor, RC 752, fabrikeres i Japan, og man skal så på fabrikken bruge en RC 702 for at kunne teste monitorerne, inden de sendes til Danmark.

Monitorerne bliver helt færdiggjort i Japan,

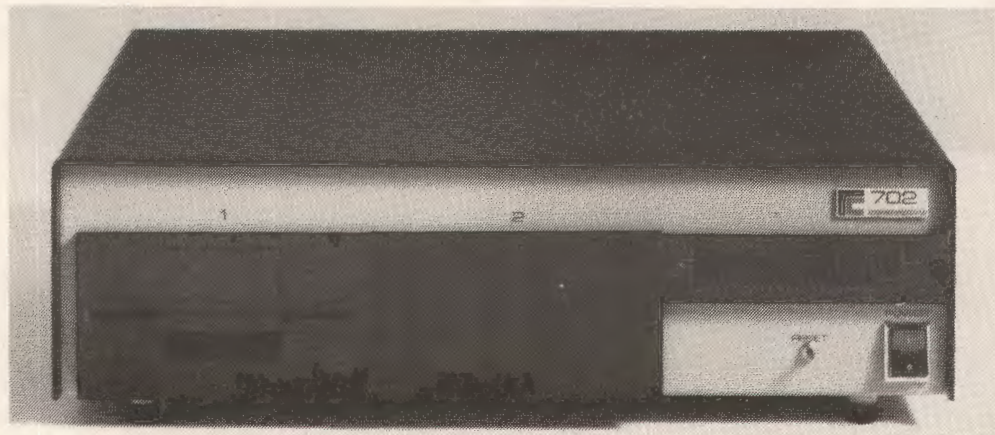
og man har haft en RC medarbejder derovre for at få alle detaljer aftalt. Monitorerne bliver leveret med RC skilte og i RC's farver og fuldt afprøvet og kontrolleret, så det vil ikke være nødvendigt at åbne kasserne, før RC 700 systemet startes op hos brugeren. Man havde flere valgmuligheder for farvekombination mellem skærm og tegn, men man besluttede tilsidst at RC 752 skulle leveres med brun skærm og gule tegn. Dette er den farvekombination, som af forskellige eksperter regnes for mest "øjvenlig", og den anbefales f. eks. af HK. Farvekombinationen kendes bl. a. også fra Alfascope terminalerne, som netop er kendt som meget brugervenlige.

Det er tanken, at RC 752 også skal kunne sælges i "løs vægt", så f. eks. de første RC 700 systemer vil kunne udskifte den gamle monitor med den nye brun/gule professionelle datamonitor.

IN OUT

I den version af COMAL, der for tiden bruges på RC700, findes sætningerne IN og OUT. De bruges til enten at hente eller aflevere et tegn på en bestemt port i maskinen. Det er f. eks. ved hjælp af OUT at man er i stand til at styre et lysbilledapparat, der tilkobles RC 700 via en lille kontrolbox, der sættes på parallel in/out porten.

Men også tastaturet på en RC 700 er tilkoblet via en port, og dette giver en række spændende muligheder: man kan nemlig lade sit program stå og spørge vedholdende på tastatur-porten, og i samme øjeblik en tast bliver



nedtrykket kan programmet gå videre, evt. styret af hvilken tast, der blev trykket på. At benytte IN på denne måde giver selvfølgelig mange muligheder for at lave spilleprogrammer og nedskyde rumskibe eller ænder; men det kan også bruges til mere seriøse ting. Hidtil er det bl. a. set anvendt til styring af et simulationsprogram for en atomreaktor, hvor man kunne trække kontrolstængerne ud eller sætte dem længere ind ved at trykke på forskellige taster. I det hele taget vil der nok være mange anvendelsesmuligheder for denne facilitet i forbindelse med simuleringer. IN er også set brugt til stavetræningsprogrammer, der f. eks. kan være udformet på den måde, at eleven har et opgaveark ved siden af maskinen med en række billeder på. Eleven skal så stave til navnene på de ting, der er vist på billederne. Fidusen ved at bruge IN i stedet for INPUT er, at man har mulighed for at kontrollere tegnene ét for ét i samme øjeblik tasterne røres. Og programmet kan så udformes på den måde, at kun rigtige tegn bliver vist på skærmen. Dette indebærer den fordel, at eleven aldrig ser et forkert ordbillede på skærmen. Trykker han på et galt bogstav sker der ikke noget. Først når han trykker rigtigt, kommer bogstavet, og maskinen går videre til næste bogstav i ordet. Selvfølgelig kan programmet "gemme" de forkerte bogstaver til senere orientering for læreren, men eleven ser aldrig et forkert stavet ord og finder altid det rigtige selv. Den helt fine udnyttelse af IN og OUT til f. eks. stavetræning kan laves på følgende måde:

En kassettebåndoptager tilsluttes via en kontrolbox lignende den, der kan styre lysbilledapparatet. På denne båndoptager er indtalt en række sætninger indeholdende passende staveord. Eleverne sidder så med hovedtelefoner på og får oplæst sætningen og det ord, der skal staves. RC 700 stopper nu båndoptageren ved hjælp af OUT og via kontrolboxen, og programmet står og venter på elevens svar via IN og tastaturet. Når dette er

klaret rigtigt, starter båndoptageren automatisk igen og går videre med næste sætning. Er man så heldig, at man kan skaffe sig en båndoptager, hvor man kan relæ-styre frem- og tilbagespoling, ja, så kan man lave de helt raffinerede programmer til undervisning, og det behøver ikke at være begrænset til stavetræning.


Sidst skal nævnes, at man også kan lade IN "fange" tryk på tasterne til cursor-styring. Dette kan bruges til en slags "simuleret lyspen" eller til "pseudo-grafisk input". Man kan nemlig lade sit program tælle antallet af berøringer af de forskellige cursor-flytningstaster, og programmet kan så regne sig til, hvor cursoren befinder sig. Og at dette kan udnyttes til mange særdeles spændende ting siger sig selv.

RC 700 NUMRE

Som ved alle andre datamatsystemer findes der også for RC 700 en lang række numre på de enkelte komponenter.

Her er en oversigt:

- RC 701 Første serie
Incl. CPU 48kb, tastatur og indbygget 8" diskette.
- RC 751 Monitor, sort/hvid (ombygget fjernsyn)
- RC 702 Nye serie
Incl. CPU 48kb, tastatur og plads til to indbyggede 5" disketter.
- RC752 Monitor, brun/gul (professionel dataskærm).
- RC791 Lineselector, 8 linier (sammenkoblingsenhed for klynger af RC 700)
- RC 761 Flexible disc 5"
- RC 762 Flexible disc 8"
- RC 861 Matrixprinter incl. semigrafik.

 **REGNECENTRALEN**
af 1979

LAUTRUPBJERG 1 — 2750 BALLERUP

Telefon: 02 65 80 00

SOFIE

Et simpelt COMAL-system til numerisk løsning af differentiaalligninger

af Keld Helsgaun

Løsning af differentiaalligninger

Differentiaalligninger optræder hyppigt ved beskrivelse af systemer. For eksempel er mange modeller af fysiske, kemiske og biologiske systemer formuleret ved hjælp af sædvanlige differentiaalligninger.

Oftentimes er det imidlertid umuligt, eller i hvert fald svært, at bestemme en *analytisk* løsning til differentiaalligningerne. For ligningen (eller mere korrekt, begyndelsesværdi-problemet)

$$dv/dt=32-v^2/8, \quad v(0)=176$$

fås, med noget besvær, løsningsfunktionen

$$v(t)=16(6+5e^{-4t})/(6-5e^{-4t})$$

Ligningen

$$dv/dt=32-v^2/8+t, \quad v(0)=176$$

har derimod ingen analytisk løsning; løsningsfunktionen $v(t)$ kan ikke beskrives ved hjælp af de matematiske standardfunktioner.

I sådanne tilfælde må man klare sig med *numeriske* løsninger, d.v.s. tilnærmede løsningsværdier for nogle udvalgte værdier af den uafhængige variabel, t .

SOFIE

SOFIE er et simpelt og effektivt COMAL-system til numerisk løsning af differentiaalligninger. Systemet foretager numerisk integration og kontrollerer automatisk fejlen.

SOFIE er let at bruge. Tag for eksempel følgende sammenhørende differentiaalligninger:

$$dv_1/dt=(-0.3+3 \cdot 10^{-7}v_2)v_1, \quad v_1(0)=1000$$

$$dv_2/dt=(0.3-0.0003v_1)v_2, \quad v_2(0)=100000$$

der er en model af konkurrencen mellem to dyrearter (Lotka-Volterras ligninger).

Et fuldstændigt SOFIE-program til numerisk løsning af disse ligninger er vist nedenfor.

```
0010 ENTER "SOFIE"
0020 LET NUMDIF=2
0030 EXEC START
0040 PROC MODEL
0050 LET D(1)=-.3+3E-07*V(2)*V(1)
0060 LET D(2)=(.3-.0003*V(1))*V(2)
0070 ENDPROC
0080 LET V(1)=1000; V(2)=100000
0090 LET TEND=30
0100 EXEC RUNMODEL
0110 PRINT "I T=";T;"ER V(1)=";V(1);"OG V(2)=";V(2)
0120 END
```

Differentiaalligningerne udtrykkes i proceduren MODEL (linje 40-70). $V(I)$ betegner værdien af den I 'te variabel, mens $D(I)$ betegner differentialkvotienten.

I starten af programmet (linje 20-30) oplyses antallet af differentiaalligninger, NUMDIF, og proceduren START kaldes (for at få dimensioneret V og D : DIM V(NUMDIF),D(NUMDIF)). Variabernes startværdier er angivet i linje 80.

I linje 90 fastsættes slutpunktet, TEND=30. Når proceduren RUNMODEL kaldes (linje 100), bestemmes en løsning til differentiaalligningssystemet i punktet $T=TEND$, idet T betegner systemets uafhængige variabel (tiden i det aktuelle økosystem).

I linje 110 udskrives sluttilstanden:

$$I T= 30 ER V(1)= 354.597 OG V(2)= 156598$$

Rapportering

Oftentimes vil brugeren imidlertid ikke nøjes med en udskrift af sluttilstanden, men ønsker for eksempel at få udskrevet modellens tilstand undervejs. Proceduren REPORT kan benyttes til at foretage en sådan løbende rapportering. REPORT skrives af brugeren og kaldes automatisk med et bestemt T -interval, REPINT.

Nedenstående program viser et eksempel på brugen af REPORT. Programmet simulerer livet i en sø. ENERGY betegner energien fra solen, $V(1)$ søens planter, $V(2)$ de planteædende dyr, $V(3)$ de kødædende dyr, $V(4)$ det organiske materiale på søbunden og $V(5)$ tabet af energi til omgivelserne.

```

0010 ENTER "SOFIE"
0020 LET NUMDIF=5
0030 EXEC START
0040 PROC MODEL
0050 LET ENERGY=95*(1+.655*SIN(2*3.14159*T))
0060 LET D(1)=ENERGY-4.03*V(1)
0070 LET D(2)=-.48*V(1)-17.87*V(2)
0080 LET D(3)=4.85*V(2)-4.65*V(3)
0090 LET D(4)=2.55*V(1)+6.12*V(2)+1.95*V(3)
0100 LET D(5)=V(1)+6.9*V(2)+2.7*V(3)
0110 ENDPROC
0120 PRINT "TIDSPUNKT","PLANTER","PLANTEÆDERE",
"KØDÆDERE"
0130 PROC REPORT
0140 PRINT T,V(1),V(2),V(3)
0150 ENDPROC
0160 LET V(1)=.83; V(2)=.003; V(3)=.0001; V(4)=0; V(5)=0
0170 REPINT=.25
0180 TEND=2
0190 EXEC RUNMODEL
0200 END

```

En kørsel med dette program gav følgende udskrift:

TIDSPUNKT	PLANTER	PLANTEÆDERE	KØDÆDERE
0	.83	.003	.0001
.25	22.3316	.484034	.190327
.5	28.4934	.766058	.558471
.75	18.3071	.558034	.651999
1	16.2769	.417932	.520334
1.25	27.9716	.678256	.55937
1.5	30.5528	.837466	.753941
1.75	19.0591	.584113	.742413
2	16.5514	.427453	.559302

I programmet er rapporteringshyppigheden, REPINT, sat til 0.25 (linje 170). Hvis REPINT ikke fastsættes af brugeren, vil værdien være negativ, hvilket betyder, at REPORT slet ikke kaldes. Hvis REPINT derimod sættes til 0, kaldes REPORT efter hvert integrationskridt.

Numerisk integration

Proceduren RUNMODEL udfører en simulering ved hjælp af en numerisk integrationsmetode, der kaldes "Fjerde (femte) ordens Runge-Kutta-Fehlberg". Ved denne metode fremskrives modeltilstanden i skridt af variabel længde. I hvert skridt estimeres integrationsfejlen, og skridtlængden justeres automatisk, så fejlen opfylder brugerens krav til nøjagtigheden.

ABSERROR og RELERROR er to variable, der kan benyttes af brugeren til at specificere henholdsvis den største absolutte og den største relative integrationsfejl, der kan accepteres. I hvert skridt må fejlen i enhver variabel V(I) højst blive

$$ABSERROR + RELERROR * ABS(V(I))$$

Ellers halveres skridtlængden, indtil fejlen kan

accepteres. Længden af et skridt, DT, må dog ikke komme under et specificeret minimum, DTMIN. Hvis det er umuligt at opnå en acceptabel fejl, selv med en skridtlængde på DTMIN, standses simuleringen med følgende fejlmeddelelse:

THE REQUESTED INTEGRATION
ACCURACY CAN NOT BE ACHIEVED

Hvis fejlen derimod er meget lille i forhold til den tilladte fejl, kan skridtlængden forøges. Længden må dog ikke overstige et specificeret maksimum, DTMAX.

Medmindre brugeren specificerer ABSERROR, RELERROR, DTMIN og DTMAX, vil følgende værdier blive benyttet:

```

ABSERROR = 0
RELERROR = 0.0001
DTMIN = (TEND-TBEGIN)/100000
DTMAX = TEND-TBEGIN

```

hvor TBEGIN og TEND betegner henholdsvis T's begyndelsesværdi og slutværdi (TBEGIN er sædvanligvis 0).

Differentialligninger af højere orden

Modellens differentialligninger skal være af første orden og kunne udtrykkes på formen

$$dv/dt = f(t,v), \quad v(0)=v_0$$

hvor v er en vektor af tilstandsvariable, og t er den uafhængige variabel i modellen.

Hvis modellen indeholder differentialligninger af højere orden, skal de omskrives til første-ordens ligninger.

En'te ordens ligning af formen

$$d^n v/dt^n = f(t,v,dv/dt,d^2v/dt^2, \dots, d^{n-1}v/dt^{n-1})$$

spaltes i de n sammenhørende første-ordens differentialligninger

$$dv_1/dt = v_2$$

$$dv_2/dt = v_3$$

:

:

:

$$dv_{n-1}/dt = v_n$$

$$dv_n/dt = f(t,v_1,v_2, \dots, v_{n-1},v_n)$$

ved at indføre nye variable:

$$v_1=v, v_2=dv/dt, v_3=d^2v/dt^2, \dots, v_n=d^{n-1}v/dt^{n-1}.$$

For eksempel omskrives anden-ordens differentialligningen (Mathieu-ligningen)

$$d^2v/dt^2 = -(a-2q \cos(2t))v$$

til følgende første-ordens system

$$dv_1/dt = v_2$$

$$dv_2/dt = -(a-2q \cos(2t))v_1$$

eller som det udtrykkes i SOFIE:

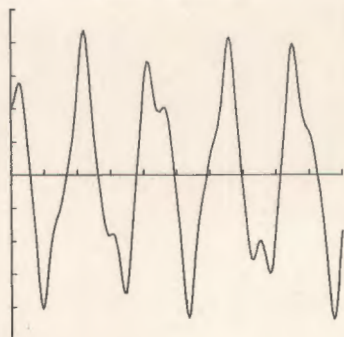
$$D(1)=V(2)$$

$$D(2)=-(A-2*Q*\text{COS}(2*T))*V(1)$$

Nedenstående program løser Mathieu-ligningen numerisk. Programmet tegner løsningskurven (T,V(1)) ved hjælp af H. B. Hansens grafiske COMAL-system (se Datalære, september 1978).

```
0010 ENTER "SOFIE"
0020 ENTER "PLOT3"
0030 LET NUMDIF=2
0040 EXEC START
0050 PROC MODEL
0060 LET D(1)=V(2)
0070 LET D(2)=-(A-2*Q*COS(2*T))*V(1)
0080 ENDPROC
0090 PROC REPORT
0100 LET X=T; Y=V(1)
0110 EXEC DRAW
0120 ENDPROC
0130 LET A=.129; Q=.55; V(1)=1; V(2)=0
0140 LET TEND=50; REPINT=0
0150 EXEC PLOTON
0160 EXEC ERASE
0170 LET XMIN=0; XMAX=TEND; YMIN=-2.5; YMAX=2.5
0180 EXEC SCALE
0190 LET X=TEND/10; Y=(YMAX-YMIN)/10
0200 EXEC AXES
0210 LET X=0; Y=V(1)
0220 EXEC MOVE
0230 EXEC RUNMODEL
0240 END
```

Resultatet af kørsel ses nedenfor.



Angivelse af en stopbetingelse

En simulering stopper, når værdien af den uafhængige variabel T bliver lig med værdien af TEND.

I SOFIE kan simuleringens ophør desuden afhænge af en bestemt betingelse. En sådan stopbetingelse specificeres i proceduren MODEL ved hjælp af variabelen STOPCOND.

Nedenstående program viser anvendelsen af STOPCOND. Programmet simulerer udskydningen af et projektil. Idet V(1) betegner projektillets højde over jorden, beregner programmet hvor længe det varer, inden projektilet rammer jorden (V(1)<0, linje 70). Landingsstidspunktet vil blive bestemt med en nøjagtighed på DTMIN.

```
0010 ENTER "SOFIE"
0020 LET NUMDIF=2
0030 EXEC START
0040 PROC MODEL
0050 LET D(1)=V(2)
0060 LET D(2)=-G-K*V(2)*ABS(V(2))
0070 LET STOPCOND=V(1)<0
0080 ENDPROC
0090 LET G=9.8; K=.01
0100 LET V(1)=0; V(2)=50
0110 LET TEND=1E+10; DTMIN=.00001; DTMAX=1
0120 EXEC RUNMODEL
0130 PRINT "TID INDEN LANDING: "; T; "SEKUNDER"
0140 END
```

En kørsel med programmet gav følgende udskrift:

TID INDEN LANDING: 7.21601 SEKUNDER

Flere simuleringer i én kørsel

Med SOFIE er det muligt at foretage flere simuleringer (kald af RUNMODEL) i samme kørsel. De følgende to eksempler viser, hvorledes denne mulighed kan udnyttes.

Det første er et eksempel på et optimeringsproblem. En servomekanisme med konstant input U giver outputtet X som bestemt ved differentialligningen

$$d^2X/dt^2 = -G(X-U) - R dX/dt, \quad X(0)=0, dX/dt(0)=0$$

Problemet består i at bestemme den værdi af dæmpningskoefficienten R, der for faste værdier for G og U giver mindst middelfvigelse mellem inputtet og outputtet, d.v.s. minimerer integralet

$$\left(\int_0^{\infty} (X-U)^2 dt \right)^{1/2}$$

Differentialligningen er af anden orden, men den kan let omskrives til et system af to førsteordens differentialligninger. Ved at benytte SOFIE's betegnelser og lade V(1)=X og V(2)=

dX/dt , og dermed $D(1)=dX/dt=V(2)$ og $D(2)=d^2X/dt^2$, kan differentialligningen omskrives til systemet

$$D(1) = V(2)$$

$$D(2) = -G*(V(1)-U)-R*V(2)$$

Middelafvigelsen kan bestemmes ved hjælp af differentialligningen

$$D(3) = (V(1)-U)^2$$

nemlig som kvadratroden af $V(3)$'s slutværdi (idet $V(3)$ er integralet af $D(3)$).

Nedenstående SOFIE-program bestemmer den værdi af dæmpningskoefficienten R , der minimerer middelafvigelsen. For at forenkle problemet antages, at R er heltallig og ligger mellem 10 og 20.

```
0010 ENTER "SOFIE"
0020 LET NUMDIF=3
0030 EXEC START
0040 PROC MODEL
0050 LET D(1)=V(2)
0060 LET D(2)=-G*(V(1)-U)-R*V(2)
0070 LET D(3)=(V(1)-U)^2
0080 ENDPROC
0090 LET V(1)=300; U=.9; V(1)=0; V(2)=0; V(3)=0; MINV3=10000
0100 LET TEND=.8; G=300; U=.9
0110 FOR R=10 TO 20
0120 EXEC RUNMODEL
0130 IF V(3)<MINV3 THEN LET MINV3=V(3); OPTR=R
0140 NEXT R
0150 PRINT "OPTIMALT R =",OPTR,"MIDDELAFGIVELSE =",SQR(MINV3)
0160 END
```

En kørsel med programmet gav følgende udskrift:

```
OPTIMALT R = 17
MIDDELAFGIVELSE = .216272
```

Det andet eksempel er en simulering af opsendelsen af en trettrinsraket fra jordens overflade. Selve modellen er beskrevet i nedenstående SOFIE-program.

```
0010 ENTER "SOFIE"
0020 LET NUMDIF=3
0030 EXEC START
0040 PROC MODEL
0050 LET D(1)=-MASSFLOW
0060 LET THRUST=MASSFLOW*FLOWVEL
0070 LET DRAG=AREA*.00119*EXP(-V(3)/24000)*V(2)^2
0080 LET GRAVITY=V(1)*32.17/(1+V(3)/2.09088E+07)^2
0090 LET D(2)=(THRUST-DRAG-GRAVITY)/V(1)
0100 LET D(3)=V(2)
0110 ENDPROC
0120 LET V(2)=0; V(3)=0
0130 REM *** FIRST STAGE ***
0140 LET V(1)=189162; MASSFLOW=930; FLOWVEL=8060; AREA=460
0150 LET TEND=150
0160 EXEC RUNMODEL
0170 REM *** SECOND STAGE ***
0180 LET V(1)=40342; MASSFLOW=81.49; FLOWVEL=13805; AREA=460
0190 LET TEND=TEND+359
0200 EXEC RUNMODEL
0210 REM *** THIRD STAGE ***
0220 LET V(1)=8137; MASSFLOW=14.75; FLOWVEL=15250; AREA=360
0230 LET TEND=TEND+479
0240 EXEC RUNMODEL
0250 END
```

$V(1)$, $V(2)$ og $V(3)$ betegner henholdsvis raketens masse, dens afstand fra jorden og dens hastighed. Under opsendelsen påvirkes raketten af de tre kræfter THRUST, DRAG og GRAVITY, hvor THRUST er brændstoffets udstødningskraft, DRAG er luftmodstanden, og GRAVITY er tyngdekraften.

De *kontinuerte* ændringer af raketens tilstand er beskrevet i proceduren MODEL (linje 40-110), hvorimod de *pludselige* ændringer, der finder sted ved frigørelsen af et rakettrin, bl. a. masse-reduktionen, er beskrevet i selve hovedprogrammet (linje 120-240). Forbrændingstiden for de tre trin er henholdsvis 150, 359 og 479 sekunder. Vi udnytter her, at RUNMODEL fortsætter simuleringen, hvis T er mindre end TEND ved kaldet.

Oversigt over SOFIE

BRUGERDEFINEREDE PROCEDURER:

MODEL:

Er beregnet til at udtrykke modellens differentialligninger, $D = f(T, V)$.

REPORT:

Er beregnet til løbende rapportering om modellens tilstand.

Rapporteringshyppigheden fastsættes ved hjælp af variabelen REPINT.

Brugervariable

NUMDIF:

Bruges til angivelse af antallet af differentialligninger i modellen. NUMDIF skal specificeres før kaldet af START.

V(I):

Hvor $1 \leq I \leq \text{NUMDIF}$, betegner værdien af den I 'te tilstandsvariabel. Værdien opdateres automatisk af RUNMODEL ved hjælp af numerisk integration af $D(I)$, men begyndelsesværdien fastsættes af brugeren.

D(I):

Hvor $1 \leq I \leq \text{NUMDIF}$, betegner differentialkvotienten for den I 'te tilstandsvariabel med hensyn til T . $D(I)$ beregnes af brugeren i proceduren MODEL.

T:

Betegner modellens uafhængige variabel. Værdien opdateres automatisk af RUNMODEL.

TBEGIN:

Bruges til at angive simuleringens startpunkt. Ved kaldet af START sættes værdien til 0.

TEND:

Bruges til at angive simuleringens slutpunkt. Ved kaldet af START sættes værdien til 0.

REPINT:

Bruges til at angive, hvor hyppigt REPORT skal kaldes.

Hvis REPINT er positiv, kaldes REPORT i punkterne TBEGIN, TBEGIN + REPINT, TBEGIN + 2 * REPINT, o.s.v.

Hvis REPINT er 0, kaldes REPORT i TBEGIN samt efter hvert integrationskridt.

Hvis derimod REPINT er negativ, kaldes REPORT slet ikke.

Ved kaldet af START sættes REPINT til -1.

DTMIN, DTMAX:

Kan benyttes til at angive henholdsvis mindste og største tilladte skridtlængde.

Hvis brugeren ikke specificerer DTMIN og DTMAX, fastsætter RUNMODEL dem således:

$$\begin{aligned}DTMIN &= (TEND-TBEGIN)/100000 \\DTMAX &= TEND-TBEGIN\end{aligned}$$

DT:

Kan benyttes til at angive længden af det første integrationskridt i en simulering.

ABSERROR, RELERROR:

Kan benyttes til at angive en øvre grænse for henholdsvis den absolutte og relative integrationsfejl, der kan accepteres i hvert skridt. Ved kaldet af START sættes ABSERROR til 0 og RELERROR til 0.0001. RUNMODEL vil automatisk justere skridtlængden, så det for hvert skridt gælder, at integrationsfejlen højst bliver

$$ABSERROR + RELERROR * ABS(V(I))$$

for samtlige tilstandsvariable $V(I)$, $1 < I < = NUMDIF$. Hvis det er umuligt at opnå den ønskede nøjagtighed, selv med den mindste tilladte skridtlængde, DTMIN, standses simuleringen med følgende fejlmeddelelse:

THE REQUESTED INTEGRATION ACCURACY
CAN NOT BE ACHIEVED

STOPCOND:

Kan benyttes i proceduren MODEL til at angive en stopbetingelse for simuleringen. Stoppunktet vil blive bestemt med en nøjagtighed på DTMIN. TEND skal stadig angives. Simuleringen ophører, når T bliver lig TEND, eller STOPCOND bliver sand (=1).

Tilgængelige brugerprocedurer

START:

Skal altid kaldes først i brugerprogrammet for at få dimensioneret V og D: DIM V(NUMDIF), D(NUMDIF).

Inden kaldet skal værdien af NUMDIF være fastsat.

RUNMODEL:

Udfører en simulering med den specificerede model (PROC MODEL), normalt fra T=BEGIN til T=TEND (se dog STOPCOND).

Ved efterfølgende kald af RUNMODEL vil

simuleringen begynde forfra (T=TBEGIN). Dog vil simuleringen fortsætte fra T til T=TEND, hvis $T < TEND$.

RESTART:

Kan benyttes af brugeren til at retablere modellens begyndelsestilstand (T=TBEGIN). (RESTART kaldes automatisk af RUNMODEL, hvis $T > = TEND$).

Implementering af SOFIE

Hele SOFIE fylder 116 COMAL-linjer. En programudskrift er vist på den følgende side.

SOFIE består af fire procedurer, nemlig tre brugerprocedurer:

RUNMODEL
START
RESTART

samt proceduren TAKESTEP (linje 9500-9546), der fungerer som hjælpeprocedure for proceduren RUNMODEL.

RUNMODEL kaldes af brugeren, når der ønskes foretaget en simulering. Proceduren udfører en simulering med T som uafhængig variabel og standser, når T bliver lig med TEND (eventuelt før, hvis STOPCOND bliver sand). Desuden sørger RUNMODEL for at kalde den brugerdefinerede procedure REPORT med den ønskede frekvens, REPINT.

RUNMODEL undersøger først, om brugeren har sat slutværdi for T, TEND, større end begyndelsesværdien, TBEGIN, og standser simuleringen med en fejlmeddelelse, hvis $TBEGIN > = TEND$ (linje 9001-9004). Derefter sørger proceduren for, at variablene DTMIN, DTMAX og DT har "passende" værdier og opfylder betingelsen $0 < = DTMIN < = DT < = DTMAX$ (linje 9006-9011). Hvis brugeren for eksempel hverken har angivet en værdi for DTMIN, DTMAX eller DT, vil de få tildelt følgende værdier:

$$\begin{aligned}DTMIN &= (TEND-TBEGIN)/100000 \\DTMAX &= TEND-TBEGIN \\DT &= (TEND-TBEGIN)/100\end{aligned}$$

I den første simulering (RUNNO=1) gemmes begyndelsesværdien for V i VBEGIN (linje 9014-9016). Dette sker med henblik på en eventuel senere retablering af V, f. eks. hvis brugeren kalder RESTART, eller hvis $T > = TEND$ ved et efterfølgende kald af RUNMODEL. Den egentlige simulering indledes med en beregning af differentialkvotienterne i begyndelsepunktet (linje 9021), og der foretages rapportering (linje 9022-9025).

Programudskrift af SOFIE

```

9000 PROC RUNMODEL
9001 IF TBEGIN>=TEND THEN
9002   PRINT "ERROR: TBEGIN>=TEND"
9003   STOP
9004 ENDIF
9005 LET ABSERROR=ABS(ABSERROR): RELERROR=ABS(RELERROR)
9006 LET DTMIN=ABS(DTMIN): DTMAX=ABS(DTMAX): DT=ABS(DT)
9007 IF DTMAX=0 THEN LET DTMAX=TEND-TBEGIN
9008 IF DT=0 THEN LET DT=DTMAX/100
9009 IF DT>DTMAX THEN LET DT=DTMAX
9010 IF DTMIN=0 THEN LET DTMIN=DT/1000
9011 IF DT<DTMIN THEN LET DTMIN=DT
9012 IF RUNNO=1 THEN
9013   LET T=TBEGIN: NEXTREP=TBEGIN
9014   FOR I=1 TO NUMDIF
9015     LET VBEGIN(I)=V(I)
9016   NEXT I
9017 ELSE
9018   IF T>=TEND THEN EXEC RESTART
9019 ENDIF
9020 LET DTFULL=DT: STOPCOND=0
9021 EXEC MODEL
9022 IF (REPINT>0 AND NEXTREP=T) OR REPINT=0 THEN
9023   EXEC REPORT
9024   LET NEXTREP=T+REPINT
9025 ENDIF
9026 IF T>=TEND OR STOPCOND THEN GOTO 9055
9027 LET TLAST=T: DT=DTFULL: TNEXT=TLAST+DT
9028 IF TNEXT>TEND THEN
9029   LET TNEXT=TEND: DT=TNEXT-TLAST
9030 ELSE
9031   IF TNEXT>NEXTREP AND REPINT>0 THEN
9032     LET TNEXT=NEXTREP: DT=TNEXT-TLAST
9033   ENDIF
9034 ENDIF
9035 FOR I=1 TO NUMDIF
9036   LET VLAST(I)=V(I): DLAST(I)=D(I)
9037 NEXT I
9038 EXEC TAKESTEP
9039 WHILE NOT ERRORACC OR (STOPCOND AND DT>DTMIN) DO
9040   IF DT<=DTMIN THEN
9041     PRINT "ERROR: THE REQUESTED INTEGRATION ACCURACY"
9042     PRINT "      CAN NOT BE ACHIEVED"
9043     STOP
9044   ENDIF
9045   LET DT=.5*DT
9046   IF DT<DTMIN THEN LET DT=DTMIN
9047   LET DTFULL=DT: TNEXT=TLAST+DT
9048   EXEC TAKESTEP
9049 ENDWHILE
9050 IF ERRORLOW AND DT<DTMAX AND DT=DTFULL THEN
9051   LET DTFULL=2*DTFULL
9052   IF DTFULL>DTMAX THEN LET DTFULL=DTMAX
9053 ENDIF
9054 GOTO 9022
9055 LET RUNNO=RUNNO+1: DT=DTFULL
9056 IF REPINT<0 THEN LET NEXTREP=TEND
9057 ENDPROC *** RUNMODEL ***

```

```

9058 PROC START
9059 DIM V(NUMDIF),D(NUMDIF),VLAST(NUMDIF),DLAST(NUMDIF),VBEGIN(NUMDIF)
9060 LET RUNNO=1: DTMIN=0: DTMAX=0: DT=0: TBEGIN=0: TEND=0: REPINT=-1
9061 LET ABSERROR=0: RELERROR=.0001: ERRORACC=1: ERRORLOW=0
9062 ENDPROC *** START ***

```

```

9063 PROC RESTART
9064 LET T=TBEGIN: NEXTREP=TBEGIN
9065 FOR I=1 TO NUMDIF
9066   LET V(I)=VBEGIN(I)
9067 NEXT I
9068 ENDPROC *** RESTART **

```

```

9500 PROC TAKESTEP
9501 DIM A1(NUMDIF),A2(NUMDIF),A3(NUMDIF),A4(NUMDIF),A5(NUMDIF)
9502 FOR I=1 TO NUMDIF
9503   LET TEMP=DT*DLAST(I): A1(I)=TEMP
9504   LET V(I)=VLAST(I)+.25*TEMP
9505 NEXT I
9506 LET T=TLAST+.25*DT
9507 EXEC MODEL
9508 FOR I=1 TO NUMDIF
9509   LET TEMP=DT*D(I): A2(I)=TEMP
9510   LET V(I)=VLAST(I)+.09375*A1(I)+.28125*TEMP
9511 NEXT I
9512 LET T=TLAST+.375*DT
9513 EXEC MODEL
9514 FOR I=1 TO NUMDIF
9515   LET TEMP=DT*D(I): A3(I)=TEMP
9516   LET V(I)=VLAST(I)+.1879381*A1(I)+.3.2772*A2(I)+.3.32089*TEMP
9517 NEXT I
9518 LET T=TLAST+.923077*DT
9519 EXEC MODEL
9520 FOR I=1 TO NUMDIF
9521   LET TEMP=DT*D(I): A4(I)=TEMP
9522   LET DV=.2.03241*A1(I)+.8*A2(I)+.7.17349*A3(I)-.205897*TEMP
9523   LET V(I)=VLAST(I)+DV
9524 NEXT I
9525 LET T=TLAST+DT
9526 EXEC MODEL
9527 FOR I=1 TO NUMDIF
9528   LET TEMP=DT*D(I): A5(I)=TEMP
9529   LET DV=-.296296*A1(I)+2*A2(I)-1.38168*A3(I)
9530   LET V(I)=VLAST(I)+DV+.452973*A4(I)-.275*TEMP
9531 NEXT I
9532 LET T=TLAST+.5*DT
9533 EXEC MODEL
9534 LET I=1: ERRORACC=1: ERRORLOW=1
9535 WHILE I<=NUMDIF AND ERRORACC DO
9536   LET DV=-2.7778E-03*A1(I)-2.89415E-02*A3(I)-2.91599E-02*A4(I)
9537   LET IERROR=DV+.02*A5(I)+3.63636E-02*DT*D(I)
9538   LET DV=-1.15741*A1(I)+.548928*A3(I)+.535331*A4(I)-.2*A5(I)
9539   LET V(I)=VLAST(I)+DV+IERROR
9540 LET TEMP=ABSERROR+RELERROR*ABS(V(I)): IERROR=ABS(IERROR)
9541 LET ERRORACC=IERROR<=TEMP: ERRORLOW=ERRORLOW AND IERROR<.02*TEMP
9542 LET I=I+1
9543 ENDWHILE
9544 LET T=TNEXT
9545 EXEC MODEL
9546 ENDPROC *** TAKESTEP: RKF 4(5) ***

```

Sidst i TAKESTEP (linje 9540-9541) undersøges integrationsfejls størrelse, IERROR. Variablen ERRORACC sættes til 1, hvis fejlen kan accepteres; ellers 0. Hvis fejlen er meget lille i forhold til brugerens krav (< 2%), kan skridtlængden muligvis fordobles, hvilket indikeres ved hjælp af variabelen ERRORLOW.

Så længe fejlen er uacceptabel, vil RUNMODEL forkaste skridtet og forsøge et nyt skridt af den gamle længde (linje 9039-9049). Skridtlængden må dog ikke blive mindre end DTMIN, og lykkes det ikke at opnå den ønskede integrationsnøjagtighed, afbrydes simuleringen med en fejlmeddelelse (linje 9040-9044).

Skridtet halveres desuden, hvis en stopbetegnelse (STOPCOND) bliver opfyldt indenfor skridtet.

Når skridtet kan accepteres, undersøges muligheden for at fordoble skridtlængden (linje 9050-9053), og der foretages eventuelt rapportering (linje 9022-9025).

Simuleringen sker i skridt ved hjælp af numerisk integration. Skridtlængden, DT, varierer mellem DTMIN og DTMAX under hensyntagen til brugerens krav til integrationsnøjagtigheden (RELERROR og ABSERROR). Skridt kortere end DTMIN kan være nødvendige for ikke at passere simuleringsslutpunktet, TEND eller det næste rapporteringspunkt, NEXTREP (linje 9028-9034). TLAST og TNEXT betegner henholdsvis begyndelsespunktet og slutpunktet for det aktuelle skridt.

Selve integrationen foretages af proceduren TAKESTEP, som ved kald tager et integrations-skridt af længde DT. Proceduren udnytter, at værdien af V og D i skridtets begyndelsespunkt, TLAST, er gemt i henholdsvis VLAST og DLAST (linje 9035-9037).

Den integrationsmetode, der er benyttet i TAKE-STEP, kaldes "Fjerde (femte) ordens Runge-Kutta-Fehlberg". Metoden anses i dag for at være en af de bedste Runge-Kutta-metoder med variabel skridtlængde. Brugeren kan skrive sin egen udgave af TAKESTEP. Som et simpelt eksempel kan tages Eulers første-ordens metode, der programmeres således:

```
9500 PROC TAKESTEP
9501 FOR I=1 TO NUMDIF
9502   V(I)=VLAST(I)+DT*DLAST(I)
9503 NEXT I
9504 T=TNEXT
9505 EXEC MODEL
9506 ENDPROC
```

Afslutning

SOFIE er et simpelt COMAL-system til kontinuert simulering. Eftersom programmerne kan fortolkes direkte, slipper brugeren for at vente på en oversættelse, hvilket er en stor fordel ved inter-

aktiv eksperimentering med modeller, især i en undervisningssituation. På den anden side kører programmerne noget langsommere end tilsvarende oversatte programmer. Det kan nævnes, at SOFIE-programmet for raket-eksemplet kørte ca. 20 sekunder på en RC3600. Samme eksempel programmeret i DISCO - et SIMULA-baseret system til kombineret kontinuert og diskret simulering, udviklet af forfatteren - kørte 0.4 sekunder på en UNIVAC 1100/80.

Interesserede kan få tilsendt SOFIE på en strimmel i ASCII-kode ved indsendelse af en adresseret og frankeret prøvekuvert til:

Keld Helsgaun
Datalogiafdelingen
Roskilde Universitetscenter
Postbox 260
4000 Roskilde

Datalærerforeningens sprogudvalg

I det sidste års tid har der ofte været formuleret ønsker om, at Datalærerforeningen burde skabe et organ, der kunne kanalisere ønsker hos brugere af undervisningssystemer.

Nu har styrelsen nedsat en sådan gruppe, der har fået medlemmerne:

Fritz G. Knudsen Styrelsen, FS
Kollerupvej 17 . 8900 Randers
06-43 49 04

Jørgen F. Hansen GS
Midgårdsvej 7 . 8600 Silkeborg
06-81 24 47

Niels O. Hansen HS
Tjørnebakken 7 . 8500 Grenå
06-32 29 81

Helge Jensen TS
Spergelvangen 2 . 8900 Randers
06-43 48 47

Børge R. Christensen COMAL-80
Viben 34 . 6270 Tønder arbejdsgruppe
04-72 26 86 FS

Gruppen har netop holdt sit første møde. Her vedtog man et arbejdsgrundlag med følgende ordlyd:

Det er udvalgets opgave at indsamle og bearbejde brugerønsker vedrørende systemer til undervisningsbrug, samt at formulere disse ønsker i offentligt tilgængelig form.

Som det fremgår, er arbejdsgrundlaget ret bredt. Dette kan også ses i de første opgaver, som gruppen har taget op: færdiggørelse af Comal-80, forhold vedrørende filbehandling, forhold vedrørende processtyring samt forhold omkring grafik. CAI-sprog og Pascal er ligeledes opgaver, der ønskes taget op til nærmere behandling.

Gruppen er selvsagt indstillet på at arbejde så udadvendt som muligt, men for at arbejdet kan få den nødvendige brede baggrund, er det vigtigt, at foreningens medlemmer henvender sig med deres ønsker.

Tøv derfor ikke med en henvendelse til gruppens medlemmer, men skriv eller ring, gerne længe inden næste møde, som finder sted 10. januar 1981.

Vedr. manuskripter til Datalære

Maskinskrevne manuskripter er velsete. Håndskrevne manuskripter må være letlæselige, og kun den ene side af papiret bør forsynes med tekst.

Ønske om bibeholdelse af afvigelser fra den "gængse" retskrivning og/eller tegn-sætning bedes angivet på manuskriptet.

VI INTRODUCERER
TIL:



**COMET
SPC-1
RC 700**

CENTRONICS MINIPRINTER 737

TEKSTBEHANDLING

80 TEGN PR. SEKUND PROPORTIONAL SPACED

50 TEGN PR. SEKUND MONO SPACED

18 x 9 MATRIX PROPORTIONAL SPACED

7 x 8 MATRIX MONO SPACED

PROPORTIONAL SPACING PLUS 10 TIL 16,5 TEGN PR. TOMME

BI-DIRECTIONAL PAPIRFREMFORING MED STEPPER MOTOR

DANSK TEGNSÆT, STORE OG SMÅ KARAKTERER

PRINT UNDERLINING

KAN SKRIVE PÅ ENDELØSE BANER, PAPIRRULLE ELLER

NORMALT PAPIR

PARALLEL : 7.495,- KR. EX. MOMS

SERIEL : 7.995,- KR. EX. MOMS



SYDVESTVEJ 53
2600 GLOSTRUP
TLF.: (02) 45 82 33

—
ADELGADE 32 B
8660 SKANDERBORG
TLF.: (06) 52 42 33

Jeg vil gerne vide mere
om CENTRONICS 737
og Interface muligheder

navn _____
adresse _____
postnr./by _____



Dansk Data Elektronik

Drejer det sig om dataanlæg til undervisningssektoren? - køb

GODT GREJ, -DER IKKE KOSTER EKSTRA!

SPC/1 BORDDATAMAT opfylder selv de største krav til en effektiv undervisning i datalogi.

De fleste anskaffer sig model SPC/1 fordi den til enhver tid kvalificerer sig blandt lignende systemer -og ikke så meget fordi

DEN ER DANSK

SPC/1 kan ligeledes anvendes i den administrative sektor. Til f.eks.:

Skemaplanlægning,
Skemaændringer,
Elevdisponeringer,
Vikartilvalg, m.v.

SPC/1 dataanlæg består af:
Skærmterminal, en printer
samt en centralenhed, der
er lager for alle infor-
mationerne.

Professionelt ville man sige, at SPC/1 er et køreklart system, der kan støtte både COMAL og Pascal multibruger-konfiguration.

dde

**Dansk Data
Elektronik ApS**
Herlev
Hovedgade 207,
2730 Herlev
02/84 50 11

