

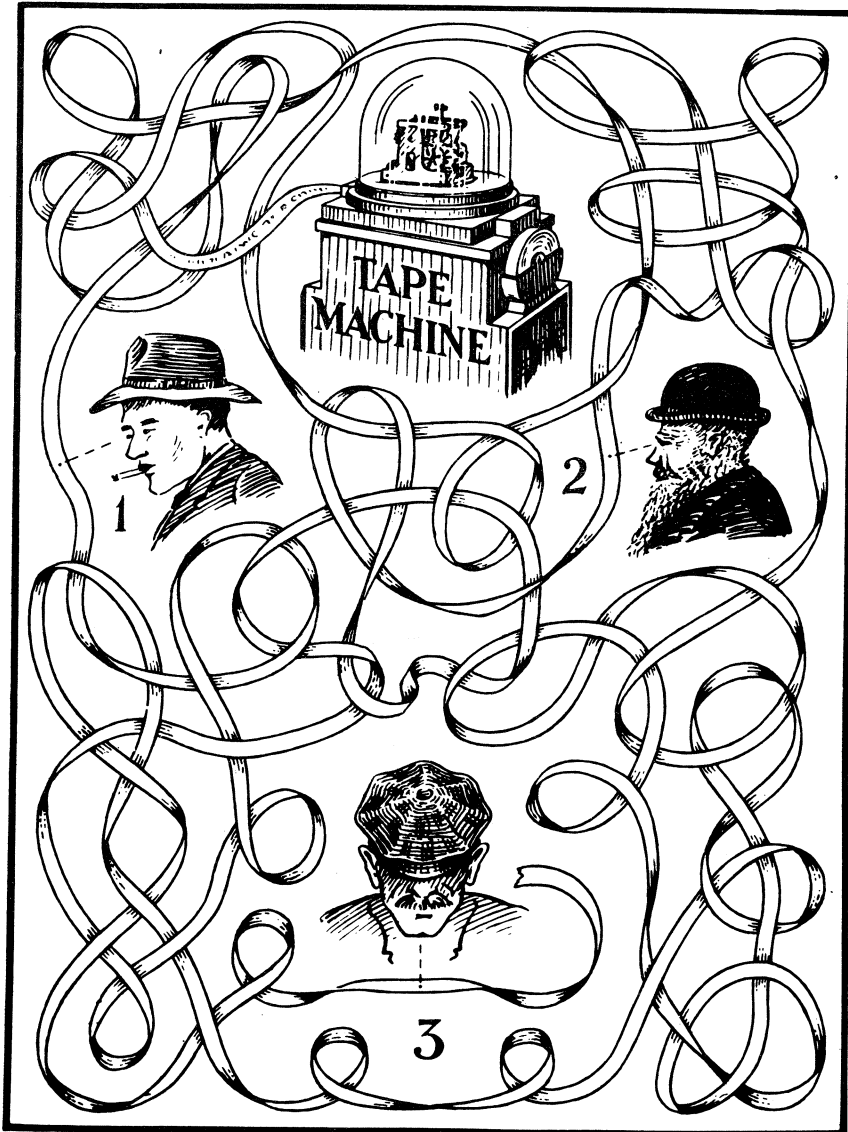
RUN · NYTT

INFORMASJONSORGAN FOR RUNIT,
REGNESENTRET VED UNIVERSITETET I TRONDHEIM

NR. 3

NR. 7

1. DES. 1980



**DATA-
ANTIKVITETER
SIDE 31**

**JULE -
NOTTER
SIDE 28**

Tre profesjonelle spillere venter på siste nytt fra veddeløpsbanen, og stirrer spent på utskriftsbåndet. Men hvem ser på riktig side av båndet? Svar side 30.

HVORDAN BRUKE UNINETT

UNINETT har nå vært i prøvedrift en tid med noen utvalgte brukere. Stabiliteten på nettet begynner nå å bli så god at vi vil la flere brukere benytte seg av nettet.

Hvordan komme i kontakt med nettet?

Pr. i dag kan nettet nås ved hjelp av terminalinngangen på RUNITs node. I løpet av året vil også terminaler som er koplet til UNIVAC 1100/60 få tilgang til nettet.

Terminalinngangene på noden kan nås ved bruk av den nylig installerte linjesvitsjen (se eget notat om bruk av linjesvitsj), eller oppringt inngang (300 b/s) på noden, telefon (9)2506.

Brukergrensesnitt

Følgende notasjon er brukt:

```
<bell> betyr control-G
<CR>   betyr vognretur (RETURN)
<LF>   betyr linjeskift
<ctr Q> hvor Q er et vilkårlig tegn,
        betyr control-Q
( )    feltet kan utelates
```

Kommandoer	Ekko	Funksjon
<ctr X>	^X<CR><LF>	Slett hele linja
<ctr Z>	-	Slett siste tegn
<ctr A>		Slå av lokalt ekko
<CR>	<CR><LF>	Avslutter linjebildet
<bell><bell>	<bell>	<bell> blir lagt i linjebildet som vanlig data.
<bell>I	<CR><LF>	lokalt avbrudd, entrer lokal kommando modus.

Lokale kommandoer

DIAL (dte),(ts),(port)	Kople opp forbindelse mot den tjenesten man ønsker å benytte.
DISC	Kople ned en forbindelse etter bruk.

Hvilke maskiner kan benyttes via nettet?

Ved universitetet i Oslo er DEC-10 tilkoplet nettet. På DEC-10 er det blant annet et konferanse-system i drift. Dette har vist seg å være praktisk i bruk dersom man jobber på et samarbeidsprosjekt med geografisk spredning.

Universitetet i Bergen har tilkopling til UNIVAC 1100/82, men har enda noen problemer med stabiliteten.

Universitetet i Tromsø vil etter planene få tilkoplet Cyber 170 til nettet i løpet av året. Forøvrig har de en prøveinstallasjon på NORD-10 tilkoplet.

I Trondheim er UNIVAC 1100/62 tilkoplet. Når programvare på NORD er ferdig, vil også NORD-maskiner bli tilknyttet nettet.

Oppkopling av samtaler

Først må lokal kommando modus entres, deretter tastes aktuell DIAL kommando.

Eks.: Oppkopling mot UI0, DEC-10
<bell>I DIAL 021005,0,3<CR>

Nettadresser:

-UI0	DTE = 021005	
	TS = 0	DEC-10
	PORT = 3	
-UIB	DTE = 021003	
	TS og PORT utelates	
-UITØ	DTE = 021001	
	TS = 1	
	PORT = 1	NORD-10
	TS = 0	
	PORT = 1	CYBER 170
-UNIT	DTE = 021002	
	TS = 1	
	PORT = 1	UNIVAC 1100/62

Det forutsettes at bruker kjenner kommando-språket til den maskina det skal kjøres mot.

Meldinger fra nettet

Ved mislykket oppkopling eller feil i nettet kan noen av følgende meldinger forekomme. Felles for alle er at samtalen er koplet ned.

-CRL OCC	Opptatt
-CRL NC	Feil i nettet som hindrer samtalen i å bli koplet opp.
-CRL RPE	Feil i nettet
-CRL NP	Anropt abonnent ikke tilgjengelig
-CRL DER	Anropt abonnent ikke tilgjengelig
-CRL DTE	Samtaler koplet ned av oppkalt abonnent.

RUN-NYTT

Redaktør : Arne Asphjell

RUNIT

7034 Trondheim-NTH

Tlf. 075 93048

Redaksjons-
assistent : Anne B. Reitan Sivertsen

Tlf. 075 93027

Utkommer : Uregelmessig foreløpig.

Abonnement: Gratis ved henvendelse til
RUNIT's ekspedisjoner eller
redaksjonen.

Bidrag : Mottas med takk!

ETTERTRYKK TILLATT NÅR KILDE ER ANGITT.

VI TRENGER

2000 TERMINALER

I det følgende oppsummerer jeg en del synspunkter på datakraftsituasjonen basert på RUNIT's siste langtidsbudsjett.

Ved inngangen til 80-årene er RUNIT særlig opptatt av

- i samarbeid med delinstitusjonene å utforme retningslinjer for fremtidig EDB-service
- utvikling av et lokalt datanett for UNIT-miljøet
- anskaffelse av flere terminaler for undervisning, forskning og administrasjon i UNIT-miljøet (delvis av brukerne, delvis av RUNIT)
- videreføring av den utbyggingslinje som startes ved overgangen til UNIVAC 1100/62 som hovedressurs
- snarest mulig å få innstallert spesielt datamaskinutstyr for større og meget beregningskrevende oppgaver
- å bedre tilgangen til datamaskinressurser og programutstyr for å tilfredsstille UNIT's behov innen administrativ databehandling
- å bygge ut småmaskinsiden med både sentralt og desentralt plassert utstyr for spesielle formål (f.eks. undervisning)

I løpet av 80-årene bør hver forsker, ingeniør etc. som jevnlig benytter edb få sin egen terminal. Likeledes vil en stor del av hjelpepersonellet ha tilsvarende utstyr. Studentene vil få dekket sitt behov i laboratorier. Basert på forsiktige vurderinger av edb-brukeren i miljøet har RUNIT kommet til et behov på ca. 1900 terminaler (mot ca. 500 i hele UNIT-miljøet i dag). "Terminaler" dekker her et vidt spektrum - fra dagens enkle terminaler til avanserte arbeidsstasjoner.

INNHOOLD:

	Side
Hvordan bruke UNINETT	2
Vi trenger 2000 terminaler	3
CTS gjør databehandling lettere	4
Ny versjon av FURPUR	6
Tilordning av filer på UNIVAC	7
Filoverføring	8
Nye programmer/programversjoner	9
Ny linjesvitsj ved RUNIT	10
OBS! Studenter!	11
Grensesnitt	12
APL	14
Mer og billigere kjøring på 1100/62	16
Slik får du vite prisen	17
Multibanking med ASCII-FORTRAN	18
Jobber som henger	19
Glemt passordet?	19
Index, tidligere artikler	20
Slik får du tilknyttet terminal	23
Kommandoer til PDP-11	24
Besøk ved Nevrologisk avd., Regionsykehuset	25
RUNITs bibliotek	26
Teknisk/matematisk gruppe	27
Test ditt hode du raske gutt	28
Svar, julenøtter	30
Bløtkakekalas	30
Hva tenker DMU-formannen?	31
Data-antikviteter	31
Data-kryss	32

forts. side 7

CTS GJØR DATABEHANDLING LETTERE

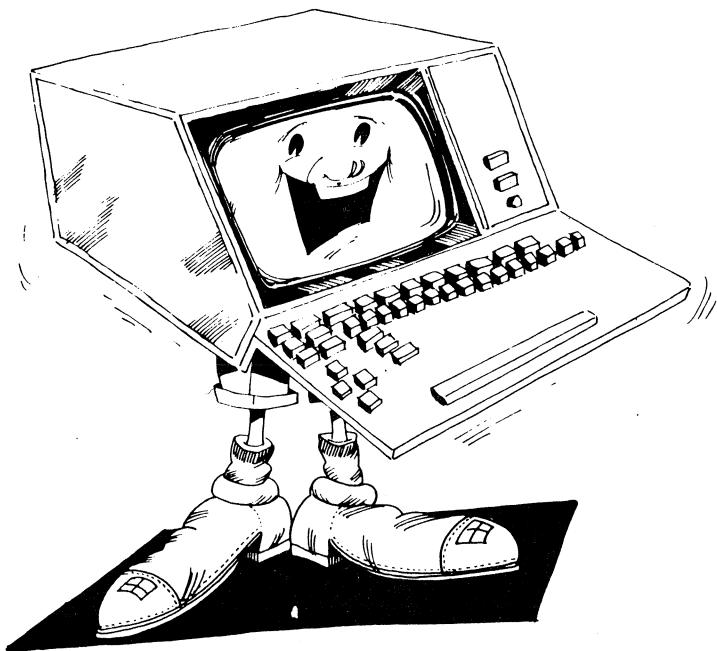
I forrige nummer av RUN-NYTT ble det gitt en kort innføring i bruk av CTS. I dette nummeret skal vi se litt nærmere på programutvikling ved hjelp av CTS.

FLERE PROGRAMMERINGSSPRÅK KAN BENYTTES

I CTS er det innebygde syntakskontrollere ("prescan" - moduler) for COBOL, BASIC, FIELDATA og ASCII-FORTRAN. Disse gir mulighet for syntakskontroll av hver programlinje etterhvert som de tastes inn fra terminal. Dersom en linje inneholder feil gis det melding om dette og brukeren får anledning til å rette opp feilen før inntasting av nye linjer fortsetter.

Når programelementet er ferdig skrevet, er det mulighet for automatisk kompilering, kollektering og utførelse av programmet. Det kan imidlertid på dette punkt være noen forskjeller mellom de ulike programmeringsspråkene, men i denne artikkelen skal vi se nærmere på bruk av ASCII-FORTRAN i forbindelse med CTS.

Det er forøvrig ingenting i veien for å benytte andre programmeringsspråk enn de som er nevnt ovenfor (f.eks. ALGOL, PASCAL). Her vil man imidlertid ikke ha mulighet for å få syntakskontroll ved inntasting av nye linjer.



HVORDAN VELGE KOMPILATOR I CTS

Standardkompilatorene kalles ved CTS-kommandoene 'COBOL', 'BASIC' eller 'FORTRAN'. Hvilken FORTRAN-kompilator som blir resultat av 'FORTRAN' - kommandoen avhenger av om CTS befinner seg i ASCII - eller FIELDATA-tilstand. Denne tilstanden velges med opsjon ved oppstartning av CTS-kjøring (q-opisjon gir ASCII, p-opisjon gir FIELDATA). Det er også mulig å skifte mellom ASCII -og FIELDATA-tilstand under en CTS-kjøring ved å bruke kommandoene 'ASSUME ASCII OFF' eller 'ASSUME ASCII ON'.

Opplysning om hvilken syntakskontroller/kompilator som for øyeblikket er aktiv kan en få ved kommandoen 'TYPE COMP()'. Brukeren merker at en syntakskontroller er aktiv bl.a. ved at '->' er byttet ut med '>>'.

Dersom man ønsker å benytte f.eks. ALGOL-kompilatoren gjøres dette ved 'ASSUME COMP ALG,S'.

Man kan komme tilbake til vanlig editor-tilstand i CTS ved å bruke kommandoen 'CLEAR'.

ASCII-FORTRAN PROGRAMMER

ASCII-FORTRAN syntakskontroller aktiveres, som før nevnt, ved kommandoen FORTRAN når CTS fra før befinner seg i ASCII-tilstand. I FORTRAN skal, som kjent, alle setninger unntatt kommentarlinjer og linjer med setningsnummer begynne i kolonne 7. Siden CTS "vet" at vi holder på med FORTRAN-programmer er det lagt inn automatisk tabulering her. Linjer som ikke begynner med 'C' eller et tall blir automatisk flyttet til kolonne 7. Dersom en programlinje (ikke kommentarlinje) skal begynne med 'C', f.eks c=100 eller CONTINUE, må vi "lure" CTS ved å slå inn en blank i starten på linja.

Ved bruk av kommandoen RUN kan vi få automatisk utførelse av det programmet vi nettopp har skrevet. Her benytter CTS seg normalt av 'checkout mode' (FTN,C) i ASCII-FORTRAN. (Se artikkel om ASCII-FORTRAN i RUN-NYTT nr.2 1980 og UNIVAC håndbok UP8244.1)..

CTS

forts.

I tillegg til automatisk tabulering vil en mulighet for å forkorte en del FORTRAN-ord bidra til å gjøre inntasting av programmer raskere. F.eks. kan CONTINUE skrives som C:E, FORMAT som F:T etc. Følgende eksempel vil forhåpentligvis illustrere programmering av enkle FORTRAN-programmer:

```
>@asg,a test.
READY
>@cts,q test.
CTS 7R1U2 10:42:04
IF YOU NEED ASSISTANCE TYPE *HELP
FOR NEW FEATURES TYPE *CALL CTS-COMMANDS
THE ASSUME MODE IS ASCII
->fortran
ASCII FORTRAN PRESCAN 1R1
>>new prog1
>>n
100 > character*60 startt,stoppt
110 >d:n a(100)
120 >sum=0
130 >startt=' program starter'
140 >stoppt=' program stopper'
150 >writ(6,61) startt
REJECTED: STATEMENT IS OF UNRECOGNIZABLE
      TYPE
>>c .t.te.
150      write(6,61) startt
160 >do 100 i=1,100
170 >read(5,50,end=200) a(i)
180 >sum=sum+a(i)
190 >100 c:e
200 >200 write(6,62) sum
210 >write(6,61) stoppt
220 >50 f:t()
230 >61 f:t(a60)
240 >62 f:t(' sum=',f10.3)
250 >end
260 >*run
FTN 9R1 *10/29/80-10:48(0,)

END FTN 28 IBANK 192 DBANK

      ENTERING USER PROGRAM
program starter
>10.0
>12.1
>9.8
>@eof
sum= 31.900
program stopper
END PROGRAM EXECUTION
>>save
>>xcts
IN EXEC MODE
```

Vi skal nå se på en noe modifisert versjon av programmet i eksemplet foran, og gjøre bruk av en rutine fra det numeriske biblioteket NAG. Dette biblioteket ligger på fila RUN*NAGFTN. I dette tilfellet kan ikke "checkout mode" benyttes da denne ikke tillater bruk av biblioteksprogrammer. Vi må derfor gå over til en ordinær kompilering (FTN,S) og kollektering av programmet. Både kompilering, kollektering og utførelse av programmet kan fortsatt skje automatisk. Endring i kallet på ASCII-FORTRAN skjer ved 'ASSUME COMP FTN,S' og angivelse av biblioteksfiler med en 'ASSUME LIBRARIES'-kommando. I tillegg til biblioteksfil for NAG må også biblioteksfil for ASCII-FORTRAN's eget bibliotek angis.

```
>@asg,a test.
READY
>@cts,q test.
CTS 7R1U2 10:56:03
IF YOU NEED ASSISTANCE TYPE *HELP
FOR NEW FEATURES TYPE *CALL CTS-COMMANDS
THE ASSUME MODE IS ASCII
->old prog1
COMPILER TYPE: FTN,C
->assume comp ftn,s
->assume lib runit*nagftn.ftn*lib
.
.
. programendringer
.
.
->print all
100      character*60 startt,stoppt
110      dimension a(100)
120      sum=0
130      startt=' program starter'
140      stoppt=' program stopper'
141      n=0
150      write(6,61) startt
160      do 100 i=1,100
170      read(5,50,end=200) a(i)
180      sum=sum+a(i)
181      n=n+1
190 100  continue
191 200  call g01aae(n,a,0,wt,xmean,
192      - s2,s3,s4,xmin,xmax,wts,ifail)
200      write(6,62) sum,xmean,xmin,xmax
210      write(6,61) stoppt
220 50  format()
230 61  format(a60)
240 62  format('sum=',f6.2,,
241      - 'middel=',f6.2,' min=',f6.2,
242      - ' max=',f6.2)
250      end
END OF FILE
```

```

->run
COMPILING
program starter
>1.0
>4.2
>8.6
>21.8
>@eof
sum= 35.60 middel= 8.90 min= 1.00
max= 21.80
program stopper
*DIAGNOSTIC SCAN? >yes
1 @FTN,S TPF$.NAME$
2 FTN 9R1 *10/29/80-11:20(0,)
25 END FTN 35 IBANK 233 DBANK
26 @PREP TPF$.
29 @MAP,S ,TPF$.NAME$
34
35 AFCM STATUS OF OUTPUT ELEMENT=UNKNOWN
36 QUARTER-WORD SENSITIVE
37
38 ADRESS LIMITS 001000 002673
.
.
.
*END DIAGNOSTIC SCAN
->@fin
IN EXEC MODE

```

Utskrift fra kompilering og kollektering styres av CTS til en egen fil. Etter utførelse av programmet får man muligheter til å se deler av denne filen ved å svare 'YES' på spørsmålet 'DIAGNOSTIC SCAN'. Det skulle gå fram av eksemplet foran. Det er også muligheter for brukeren selv når det måtte ønskes å se på hele eller deler av denne filen ved å benytte kommandoen SCAN. Dette vil imidlertid ikke bli behandlet nærmere her.

Hvis kun kompilering av programmer ønskes kan kommandoen 'COMP' benyttes istedet for RUN. Det relokerbare element som blir resultat av 'COMP'-kommandoen legges normalt på fila TPF\$. Det samme gjelder det absoluttprogrammet som blir laget ved bruk av 'RUN'-kommandoen. Relokerbare- og absoluttprogrammer kan styres til en egen fil ved bruk av kommandoen 'ASSUME OBJECT filnavn'. Hvis vi vil sette egne navn på relokerbare- eller absolutt-programmer kan dette gjøres ved bruk av h.h.v. 'ASSUME REL relnavn' og 'ASSUME XQT absnavn'.

Paul Rusten

Ny versjon av FURPUR

Versjon 28R1 av FURPUR vil bli tatt i bruk, og dette vil medføre en del endringer for brukerne:

@PACK programfil.:

Hvis innholdsfortegnelsen (TOC) på en fil viser at to elementer overlapper hverandre vil dette nå bli oppdaget. Det vil bli skrevet ut en melding og pakkingen av fila blir avsluttet. Fila kan være helt eller delvis ødelagt, men det kan være mulig å kopiere deler av denne over på en ny fil.

@COPY,I fil1.elt1,fil2.:

COPY,I er nå utvidet slik at det vil bli mulig å kopiere fra element til fil.

@PREP programfil.:

Hvis en entrepunkttabell allerede eksisterer vil ikke PREP lage en ny. PREP,F vil imidlertid medføre at entrepunkt-tabell blir produsert selv om det finnes en entrepunkt-tabell på fila fra før.

@ERS,Z fil.:

Z-opsjonen vil sørge for at innholdet på fila blir overskrevet før plassen blir frigjort.

@ERS,D fil.:

Innholdet på fila blir overskrevet og deretter blir fila slettet.

@COPY,UD fil1.,fil2.:

U-opsjonen i forrige versjon sørger for at fil2 med neste ledige syklusnummer, fil2(+1), blir opprettet og at fil1 inklusive lese- og skrivenøkler blir kopiert over til denne. I tillegg blir forrige filsyklus av fil2 slettet. Nytt er D-opsjon, som forhindrer at forrige filsyklus blir slettet.

@COPY,G :

Ved kopiering fra masselagerfil til magnetbånd vil M-opsjon automatisk bli satt inn dersom dette ikke er gjort av bruker. Brukere som idag benytter @COPY,G og deretter @MARK bør endre dette til @COPY,GM for å unngå problemene senere.

GENERELT:

Flere prosessorer (f.eks. @PREP og @PACK) vil skrive ut mer informasjon enn før. For flere detaljer henvises det til UNIVAC-håndboka UP-4144, vol 3 (PRM).

Paul Rusten.

SLIK SKAL DU TILORDNE FILER PÅ UNIVAC

Den mest effektive måten å tilordne filer på, er å plassere alle tilordningssetningene (@ASG) helt først i kjøringen, like etter RUN-kortet. Årsaken er at operativsystemet foretar en såkalt "initial facility scan", dvs. ser gjennom alle tilordninger av filer først i jobben, før jobben åpnes. Når et kontrollkort som ikke er @ASG, @CAT, @USE, @FREE, @HDG, @MSG, @LOG, @EOF eller @QUAL påtreffes, avsluttes søkingen og de påtrufne ASG-kort behandles før de etterfølgende kontrollkort leses. Jobben blir derfor i praksis ikke åpnet før det er sjekket at alle filtilordningene kan tilfredstilles.

Hvis flere filer som skal brukes i kjøringen er lastet ut, vil systemet starte en kjøring som laster inn alle disse før selve kjøringen starter på denne måten. Det er mye mer effektivt enn å foreta alle enkle innlastinger med jevne mellomrom utover kjøringen. I det siste tilfellet øker gjennomløpstiden både for den aktuelle jobben og andre jobber ved at den legger beslag på operativsystemet.

En annen fordel er at jobben på denne måten heller ikke opptar plass blant de satsvise jobber som til enhver tid kan være åpne samtidig. Eksemplet kan sammenlignes med en butikkunde som venter med å fylle ut sjekken til han kommer til kassen.

Eksempler på effektiv og ineffektiv måte å gjøre tilordninger på:

Ineffektiv:

```
@RUN,/TP
@ASG,A A.
@SYM PRINT$,,PR
@ASG,A B.
.
.
@FIN
```

Effektiv:

```
@RUN,/TP
@ASG,A A.
@ASG,A B.
@SYM PRINT$,,PR
@SYM PRINT$,,PR
.
```

Første eksempel: Dersom første fil er lastet ut vil jobben passiviseres til filen er lastet inn. Deretter behandles

SYM-kortet og neste ASG-kort. Hvis også neste fil er lastet ut, må en ny kjøring settes igang for også å få inn denne.

I det andre tilfellet vil filene, hvis de er lastet ut, bli tatt inn igjen før jobben åpnes.

En mer utførlig beskrivelse av denne mekanismen finnes i Programmers Reference Manual (UP-4144.21) vol.2, kap. 12 side 9-10.

LEDER forts.

Behovet for behandlings- og lagringskapasitet vil bli dekket av lokale småmaskiner, lokale og sentrale spesialmaskiner og sentrale generelle maskiner. En vekst i behovet proposjonalt med veksten i terminalantallet gir 4x dagens nivå, men det er ventet at behovet vil vokse raskere.

De fleste datamaskinene og terminalene vil være bundet sammen vha. et lokalt datanett basert på standardiserte tilknytningsformer. Dagens nett må moderniseres og bygges vesentlig ut.

De siste årene har de bevilgninger RUNIT har fått til edb-utstyr (m. programvare) ligget 0.5-1.5 mill. under behovet, noe som sterkt har hemmet en naturlig utvikling.

Den gunstige prisutviklingen for edb-utstyr fra 70-årene kan neppe ventes å fortsette i 80-årene - bl.a. pga. kostnadene for programvaren. Vi venter at behovet for edb-kapasitet - slik det er beskrevet foran - vil øke raskere enn prisene synker. Dette gir et økende bevilgningsbehov.

RUNIT's LTB tilsier at det er nødvendig med ca. 7 mill. kr. årlig bevilget fra UNIT til edb-utstyr, mot 4.5 mill. i 1980. I tillegg kommer de bevilgninger som naturlig gis direkte til brukermiljøene, især til utstyr for den enkelte arbeidsplass (kr. 15-70.000 avhengig av funksjoner m.m.). Total investering i nødvendig edb-utstyr kan dermed komme opp i 12-14 mill.kr. årlig.

Edb blir enda mere sentralt i de fleste fag i 80-årene enn tidligere. I tillegg kommer også behovene innen administrasjonen og bibliotekene.

Skal edb-tjenestene i UNIT-miljøet dekke det forventede behov i 80-årene både i størrelse og bredde, må bevilgningene til edb-utstyr økes kraftig allerede fra 1981/82.

Karl G. Schjetne

FILOVERFØRING

Da det har vært en del misforståelser angående filoverføring mellom NORD/UNIVAC og NORD/NORD vil vi her komme med en forhåpentligvis oppklarende oversikt.

UNIVAC → NORD

- UNIVAC 1100/60 har forbindelse med NORD 100. UNIVAC 1100/21 har forbindelse med NORD 10/A.

- Fila som skal overføres må være ei datafil eller et element i ei programfil. M.a.o., for å kunne overføre hele elementfiler må man ta element for element.

- Overføringa skjer med et prosessorkall slik:

UNIVAC 1100/21:

```
@E8*D.N10 filnavn.elementnavn,
brukernavn-nord*filnavn-nord.
filtype-nord,F17CPA (eller N10A7)
```

UNIVAC 1100/60:

```
@E8*D.N100 filnavn.elementnavn,
brukernavn-nord*filnavn-nord.
filtype-nord,F09CPA
```

eks.:

```
@E8*D.N10 UNIVACFIL.ELEMT,USER*NORDFIL.SYM,
F17CPA
```

UNIVACFIL.ELEMT	er navn på fil og element på UNIVAC 1100.
NORDFIL:SYM	er navn på fil og filtype på NORD.
USER	er brukernavn på NORD. NORD-fila må ha PUBLIC-ACCESS = WA (write,append).

Filaksessen settes ved kommandoen

```
@SET-FILE-ACCESS <filnavn>
```

```
PUBLIC ACCESS -----: WA
FRIEND ACCESS -----: --
OWN ACCESS -----: --
```

NORD → UNIVAC

- Fila som skal overføres må være symbolsk
- Fila må ha PUBLIC ACCESS = R
- Fila må være oppbygd som en satsvis jobb på UNIVAC

```
eks.: @RUN <brukernr>,<kjørenr>,,4,200
@PASSWD <passord>
@SYM PRINT$,,<printer>
@ELT,ID <qualifier>*<filnavn>
.<elementnavn>
.
.
.
fil fra NORD
.
.
.
@END
@FIN
```

OBS:

UNIVAC-styresetninger må legges inn v.h.a. editor (QED,PED el.l.)

- Etter fila er klargjort for overføring skriver man SINTRAN-kommandoen:
APPEND-REMOTE UNIVAC "filnavn på NORD"

Det finnes en brukerbeskrivelse for filoverføring både på UNIVAC og NORD ved navn:

```
UNIVAC: 1100/60 - E8*D.N100/DOC
1100/21 - E8*D.N10/DOC
NORD: 100 - (UNIVAC)N100DOC:SYMB
```

NORD ↔ NORD

Overføring mellom NORD-maskinene er mulig via en såkalt SINTRAN-SINTRAN link. Denne linken består av 4 kanaler, hvorav to brukes til filoverføring, disse kalles CHANNEL-3 og CHANNEL-4.

Eksempel på overføring fra REMOTE maskin til LOCAL maskin:

- Logg inn på vanlig måte ved LOCAL maskin

- Videre:

```
@REMOTE 1
CHANNEL NUMBERS: LOCAL - 600, REMOTE - 600
```

FILOVERFØRING forts.

Logg inn på REMOTE maskin
 <ESC>

ENTER <brukernavn>
 PASSWORD:
 OK
 PROSJECT NUMBER:

R@
 Her angir R at du er på REMOTE maskin.

Trykk eller <RUB OUT> tast og du er tilbake på LOCAL maskin og forbindelse med REMOTE maskin er opprettet.

@COPY-FILE "destination-file" CH-3. source file eller CH-4.source file

(Ved kopiering fra LOCAL til REMOTE ser det slik ut

@COPY-FILE CH-3."destination file" source file eller CH-4."destination file" source file)

Når kopiering er ferdig avslutter man REMOTE forbindelsen slik:

```
@REMOTE      over til REMOTE maskin
R@LOG        logg ut
---
---
--EXIT--
<DEL>        Trykk<DEL> eller<RUB OUT>
@LOCAL
@            forbindelsen er brutt
```

NB! For systemts stabilitet er det meget viktig at forbindelsen mot REMOTE-maskinen avsluttes umiddelbart som vist ovenfor.

Brit Bonsaksen/Rolf Westly



NYE PROGRAMMER/ PROGRAMVERSJONER

COBOL

Versjon 5R1 av ASCII-COBOL kan kalles ved
 @RUNIT*NEW.ACOB
 Biblioteksfil er NEW*ACOB.

SIMULA

Versjon 3R5 av Simula vil om kort tid erstatte versjon 3R4. Kall blir som nå:

```
@RUNIT*NEW.SIM
eller
@NEW*SIM.SIM
```

Det siste kallet må benyttes dersom vi ønsker automatisk kollektering og utførelse. Biblioteksfil er fortsatt NEW*SIM.

PASCAL

En PASCAL-kompilator er nå under uttesting på UNIVAC. Den kan kalles ved:

```
@PASCAL
Biblioteksfil er PAS*LIB. Mer informasjon om PASCAL-kompilatoren på UNIVAC vil bli gitt på et senere tidspunkt.
```

SPSS

Versjon 8.1-UW1.0 vil om kort tid bli lagt inn og kan kalles ved:

```
@RUNIT*SPSS.SPSS
Den nye versjonen er en forbedret utgave av den forrige versjonen og inneholder ingen vesentlige nyheter.
```

FURPUR

Versjon 28R1 av FURPUR er lagt inn. Informasjon om den nye FURPUR-versjonen finnes et annet sted i dette nr. av RUN-NYTT.

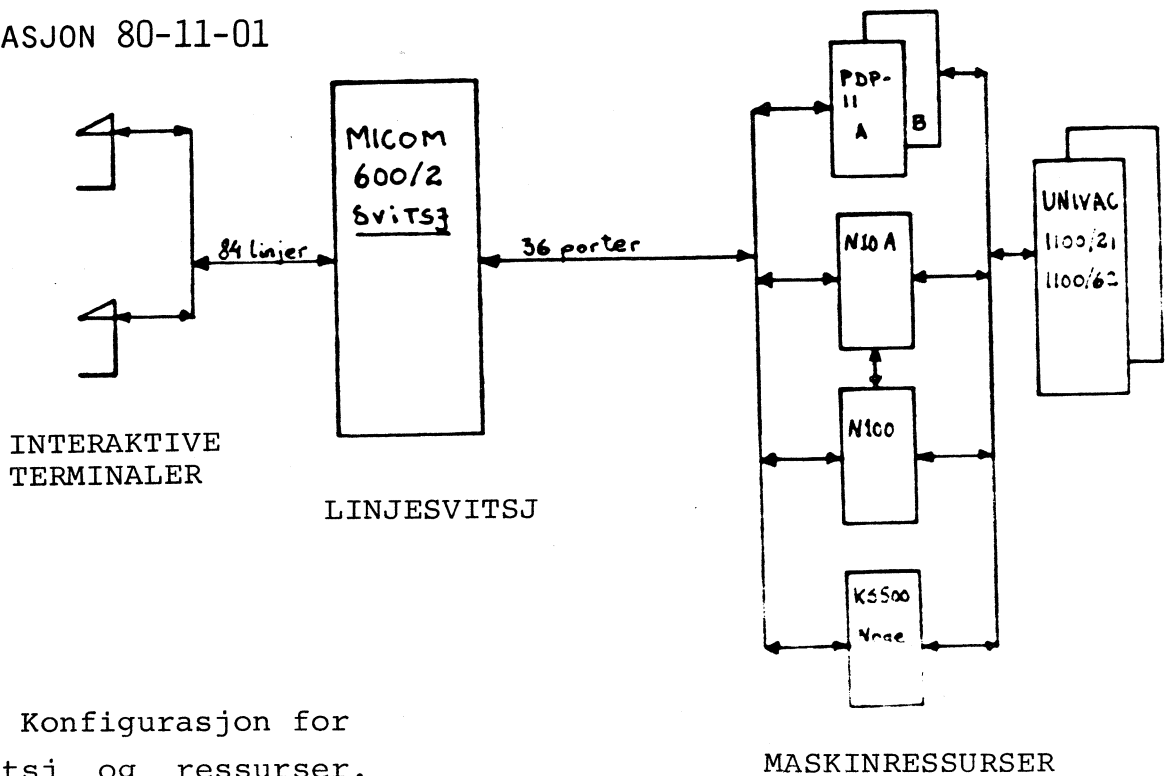
Henvendelse angående nye programversjoner rettes til RUNITs brukerkontakt- og programvaregruppe tlf. (9)3029.

NY LINJESVITSJ VED RUNIT

Antallet terminaler i UNIT- miljøet har vokst meget sterkt i den senere tid. De fleste er knyttet til datautstyret (UNITS hoveddataanlegg) via faste trådsamband, og med en gjennomsnittlig lav aktivitetsprosent blir utnyttelsen av maskininngangene dårlig.

I tillegg til at nye maskininnganger koster penger, har hver enkelt maskin begrensninger på hvor mange terminalinnganger den har plass til totalt, og hvor mange av disse som kan være simultant aktive.

KONFIGURASJON 80-11-01



Figur 1. Konfigurasjon for linjesvitsj og ressurser.

UTSTYRSKONFIGURASJON

Figur 1 viser hvordan linjesvitsjen passer inn i bildet sammen med de øvrige ressursene som tilbys. Svitsjen er slik konfigurert at alle terminalene kan nå alle maskinressursene forutsatt at det er en ledig inngang i øyeblikket, og at terminalens hastighet korresponderer med maskininngangens. I tillegg må brukeren være registrert (passord o.l.) på den maskinen han/hun ønsker å kjøre mot.

For de samband som går via linjesvitsjen er terminalinngangen på svitsjen som en "linje", og utgangen fra svitsjen til data-maskininngangen som en "port".

For å bøte på dette kan vi innføre et "sentralbord" som tildeler terminalbrukerne innganger etter prinsippet "først i køen blir først betjent". En annen betegnelse på et slikt "sentralbord" er linjesvitsj.

Høsten 1979 ble det bestemt å finne ut om det fantes passende utstyr på markedet for å løse våre behov for en bedre utnyttelse av datamaskinressursene. Denne undersøkelsen resulterte i en inngående studie av to aktuelle linjesvitsjer, og valget falt til slutt på en MICOM modell 600/2.

Når svitsjen blir satt i drift (ca.80-11-25) vil noen av terminalene fra Sentralbygg II bli koplet opp mot svitsjen, og fra disse vil en kunne velge å kjøre mot NORD-10A, NORD-100, UNIVAC (PDP-11) og KS-500 datanett-node. Maskininngangene (portene) vil fordele seg med et passende antall i hver maskin med hastigheter på 300, 1200 og 4800 bps (biter pr. sekund). Antall innganger på svitsjen (linjer) er 84, og antall utganger (porter) er 36. I det samme kabinettet kan svitsjen ha 496 innganger (linjer) og utganger (porter) til sammen (sum linjer/porter). Fullt utbygget (2 kabinetter) blir det 992 linjer/porter.

Etter hvert som nye terminaler blir flyttet over fra de enkelte maskinressursene til linjesvitsjen vil dette kunne frigjøre maskininnganger. Disse kan så disponeres til nye brukere.

NY LINJESVITSJ **forts.**

INN OG UTLOGGINGSSEKVENSER, TJENESTER

Som fig. 2 viser, er de forskjellige maskinressurser og hastigheter definert som klasser. For å kunne kommunisere med en bestemt maskin, velger en først en klasse. Er det ledige porter med riktig klasse, vil svitsjen koble til den første ledige. Er derimot alle opptatt kan en velge om en vil stå i ventekø eller ikke. Etter å ha fått klarsignal, er svitsjen helt transparent, og en kan logge seg inn på den vanlige måten.

KOKEBOK FOR INNLOGGING

1. Velg den samme hastighet på terminalen som den klassen du er interessert i.
2. Hvis du har valgt 4800 bps, trykk SPACE. Hvis andre hastigheter er valgt, trykk CR.
3. Er riktig karakter trykket, vil svitsjen svare:.
"ENTER CLASS".
Har en derimot trykket på feil karakter, vil svaret en får bare være tull. Vent ca. 8-10 sekunder til du får enda et svar, bare tull det også. Trykk deretter på riktig karakter.
1. Etter at "ENTER CLASS" er kommet, velg klasse etter tabellen i fig. 2 innen 12 sekunder.
5. Hvis det finnes en ledig port med valgt klasse, svarer svitsjen med "GO" og en er koblet til valgt ressurs.

Er derimot alle porter i klassen opptatt vil svitsjen svare med "BUSY, WAIT?nnn". "nnn" angir koeffisienten. Vil en vente skriver en Y, hvis ikke N.

UTLOGGING

Det er viktig at en etter å ha logget ut jobben, også logger ut fra svitsjen. Gjør en ikke det, vil snart alle portene være opptatt. Utloggingen foregår ved hjelp av 3 BREAK etter hverandre, hver på ca 1 sekund.

MELDINGER

Følgende meldinger kan dukke opp ved forsøk på logging:

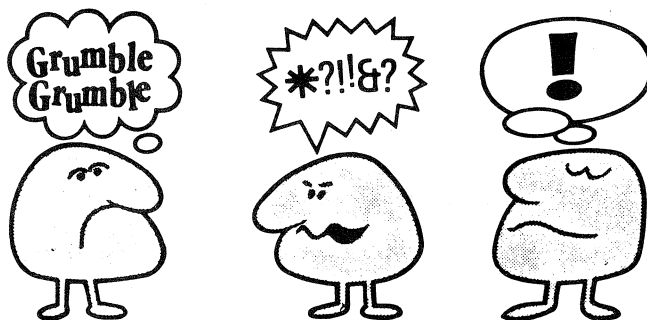
"*DISCONNECTED": En er koplet fra svitsjen, og må begynne på innloggingen.

"UNASSIGNED": Klassen som er valgt eksisterer ikke.

"UNAVAILABLE" Klassen er ute av drift.

Ressurser	Hastigheter:
01x:UNIVAC (PDP-11)	xx1: 300 bps
02x:NORD-10A	xx2:1200 bps
03x:NORD-100	xx3:4800 bps
04x:KS-500 datanett-node	
Eks. Class = 023 NORD-10A, 4800 bps.	

Figur 2.
Ressurser og hastigheter identifisert i klasser.



OBS! STUDENTER!

Alle som har typekode A i kjørenummeret må ha brukernummeret i RUN--kortets prosjektfelt! Hvis ikke blir det problemer i forbindelse med filtilordning. Eksempel på korrekt RUN-kort:

@RUN,/TP 22222,CA8001A,22222,S10,20

GRENSESNIITT

NOTIS-1 TEKSTBEHANDLINGSSYSTEM

Alle større dokumenter, så som rapporter, prosjektarbeider og diplomoppgaver kan med fordel databehandles fremfor å maskin-skrives. Ulempene kan sammenfattes til at en må bruke et par timer på å sette seg inn i et nytt system, og at en trenger tilgang til en dataskjerm. Fordelene består i sterkt forbedrede muligheter til å rette feil og å sette inn nytt stoff der det måtte være behov for det, til å holde orden på dokumentene, ved hjelp av automatisk generering av innholdsfortegnelse og indeks, og til lett omredigering. Dessuten får en penere utseende på dokumentet, fordi en får rett høyremarg.

Undertegnede har spesiell erfaring med NOTIS-1 på NORD datamaskiner. Systemet omfatter en skjermorientert teksteditor (TED), et tekstredigeringsprogram (TEXT-FORMATTER eller TEKST-FORMATERER) og et utskriftsprogram (SCAN).

Det er spesielt tilgangen til en skjerm-editor som letter skrivearbeidet så enormt. Det er like raskt å rette skrivefeil som å skrive ny tekst, og det er lett å lage tabeller og enkle grafer. (Mer kompliserte figurer kan en tegne ut for hånd, eller ved hjelp av en datastyrt plotter, og klebes inn senere). Retting av skrivefeil, og spesielt oppsett av tabeller er som kjent nesten håpløst arbeid hvis en benytter en skrivemaskin, og fremdeles svært arbeids og tidkrevende hvis en benytter en linjeorientert editor. TED omfatter alle vanlige editorfunksjoner som innsetting og sletting av linjer eller deler av linjer, flytting og duplisering av tekst, men også mer avanserte funksjoner som splitting og sammen-setting av linjer, og kopiering av tekst fra linjen over eller under. Det som er så spesielt med skjermeditoren er at det er uhyre raskt å skifte fra linje til linje, og å posisjonere seg på linjen, bare ved å flytte en markør (cursor) rundt på skjermen. Bildet ruller automatisk opp eller ned på skjermen hvis en flytter markøren utenfor bildet. Det tar bare ca. en halv time å venne seg til skjermeditoren hvis en i det hele tatt har brukt en editor før, og en kan da som regel ikke tenke seg å gå tilbake til en linjeeditor igjen.

Tekstredigeringsprogrammet er også meget enkelt å bruke. En legger inn direktiver (redigeringskommandoer) i teksten ved hjelp av editoren, og kaller så opp redigeringsprogrammet direkte. En klarer seg lenge med å kunne ca. 10 av de vanligste direktivene, og de som har mer avansert behov har lært det meste hvis de bruker et par timer på å lese gjennom brukerveiledningen (som er til salgs på TAPIR). Direktivreportoaret omfatter også makroer, dvs. navngitte flerkommandosekvenser der det også kan legges inn standardverdier på variable. Makroene kalles på samme måte som andre kommandoer.

De mest brukte direktivene er:

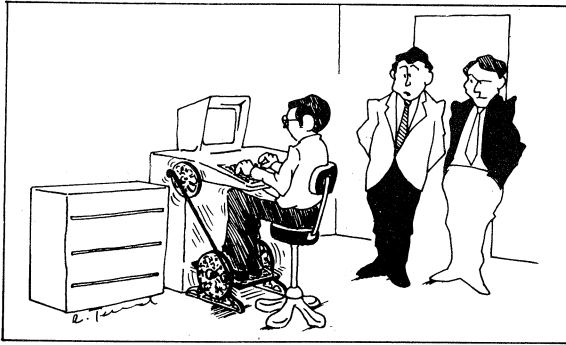
- ⌘P; gir et nytt avsnitt (Paragraph) i teksten
- ⌘CH, kapittelnavn; definerer navn på et kapittel slik at det blir skrevet ut både i teksten og innholdsfortegnelsen
- ⌘PG; gir sideskift (PaGe).
- ⌘SD; definerer navngitte seksjoner og
- ⌘SE; underseksjoner innen hvert kapittel.
- ⌘SU; pittel.
- ⌘TI; dokumenttittel.
- ⌘TP; skriver ut innholdsfortegnelse og en tittelside med den dokumenttittelen som defineres med TI.
- ⌘A; start Nyttige direktiver når en skal
- ⌘; slutt lage tabeller da de definerer absolutt-modus-områder, dvs. de blir ikke omredigert av redigeringsprogrammet.

På UNIVAC-anlegget er det ikke tilgjengelig skjermterminal for vanlige brukere. Dessuten skriver de fleste linjeskriverne på UNIVAC utelukkende store bokstaver. Det samme gjelder de fleste skjermterminalene tilknyttet den maskinen. Disse forholdene gjør det ganske håpløst å benytte UNIVAC's tekstbehandlingssystem, @DOC.

Per Agnar Sollid
Stud. techn.

P.S. Kan "rette vedkommende" innen RUNIT opplyse om hvilke utvidelser av tekstbehandlingmulighetene RUNIT har tenkt å tilby i tiden fremover?





'In Many Ways It's a Very Advanced System.'

RUNITs tekstbehandlingspolitikk

Spørsmål som det vi får her er ikke særlig morsomme, fordi RUNIT ikke har noen brukervennlig tekstbehandlingspolitikk i øyeblikket. Men for å summere opp noen hovedpunkter:

* RUNIT har et prosjekt gående med det ambisiøse navnet "Morgendagens kontor". Den første fase av dette prosjektet går ut på å avklare hvordan RUNIT internt skal dekke sine tekstbehandlingsoppgaver. Her er NOTIS et av de mest aktuelle alternativer.

* RUNIT har ikke gått ut og offisielt tilbudt tekstbehandlingstjenester på sine datamaskinressurser. Årsaken er delvis at ED/DOC på UNIVAC-maskinene er litt for dataorientert til å være brukervennlig nok for nybegynnere, dels at det ikke har vært mulig å tilby utskriftsmuligheter av tilstrekkelig kvalitet. ASCII-skriveren gir ikke god nok kvalitet for reproduksjon, og dokument skrivere av Diablo-type er både for sene og lunefulle til at de egner seg for masseutskrifter.

* Et tekstbehandlingsutvalg i UNIT-regi er i arbeid for å komme med en innstilling om hvordan universitetet som helhet bør dekke sine tekstbehandlingsbehov. En avklaring av dette vil komme i løpet av 1981, men sannsynligvis vil det gå i retning av et større sentralisert anlegg med mindre lokale anlegg tilknyttet, og med RUNIT som ansvarlig for driften av de sentrale ressursene.

I mellomtiden er det selvsagt ingenting i veien for at brukerne benytter seg av ED/DOC på UNIVAC eller NOTIS på NORD-100 maskinen (begrenset adgang for studenter). Selv om det i SBII finnes terminaler tilknyttet UNIVAC som er tilgjengelig for "offentligheten" (også en ASCII-terminal), skal det villig innrømmes at på grunn av begrensede inn/utmuligheter er RUNITs tekstbehandlingstilbud dårlig i øyeblikket.

Arne Asphjell

@ED - kommandoen END.

Det er ofte lett å glemme å gå over til INPUT - modus under editering av programmer på UNIVAC. Dette er som regel greit å rette på, unntatt i ETT tilfelle:

Dersom den inntastede linje består av ordet END er konsekvensen at UNIVAC forlater editor med følgende opplysning:

NO CORRECTIONS APPLIED.

Og DET var jo ikke meningen.....

Ingrid Selseth og Else Lervik.

P.S. Er det mulig å fjerne denne kommandoen fra editor, eller har den noen praktisk nytte?

Det finnes andre kommandoer som kan gi samme effekt. F.eks. hvis EXIT og OMIT skal legges inn som tekst og man ikke befinner seg i "INPUT" - tilstand. I de aller fleste tilfelle vil man imidlertid få en feilmelding.

Det er neppe noen løsning å forandre editoren. For det første er det ikke sikkert at alle vil være tilfreds med slike endringer. For det andre tror vi ikke at dette problemet er så stort at det kan forsvares å lage en lokal endring av editoren. RUNIT har som holdning at vi i størst mulig utstrekning skal unngå å endre standardprogrammer fra leverandør. Problemet er ikke bare å utføre forandringen en gang, men det må også følges opp og holdes ved like ettersom vi mottar nye versjoner av programmene.

Paul Rusten



'We'd Never Keep Our Accounts Straight Without It.'

APL

I RUN-NYTT nr. 2, 1980, side 12 ble programmeringsspråket APL presentert. I dette nummeret vises eksempler på kommandoer og operatorer i APL.

På UNIVAC 1100/62 kalles APL på følgende måte:

a) Fra APL-terminal, dvs. terminal med APL-spesialtegn:

```
@APL,T
```

b) Fra vanlig terminal:

```
@APL
```

Da må en bruke såkalte "digraphs" for de forskjellige APL-tegnene. "Digraphs" består av ett eller to vanlige terminaltegn.

En kommer inn i kalkulasjonstilstand, og kan direkte gjøre beregninger interaktivt med de ulike operatorene. APL har også en definisjonstilstand for skriving av program.

APL prosessoren arbeider mot et såkalt aktivt arbeidsområde. Innholdet i dette arbeidsområdet kan ved hjelp av visse APL-kommandoer lagres for senere bruk på en spesiell APL lagerfil. På denne filen lagres hvert arbeidsområde v.h.a. et biblioteksnummer som utregnes av APL systemet ut fra brukernummer, kjørenummer og prosjektfelt på RUN-kortet.

Brukeren må derfor logge seg inn på maskina på samme måte når han senere skal ha tak i det som er lagret.

APL kommandoer

APL kommandoer brukes til å styre APL prosessoren. Alle kommandoer starter med).

)SAVE LAGR Innholdet i det aktive arbeidsområdet lagres på APL lagerfilen med navn LAGR.

)LOAD LAGR Innlesing av element LAGR på lagerfilen til det aktive arbeidsområdet. Innholdet i arbeidsområdet overskrives.

)COPY LAGR Innlesing av LAGR som legges til innholdet i arbeidsområdet

)LIB Forteller navnet på de elementene brukeren har lagret på lagerfilen under sitt biblioteksnummer.

)DROP LAGR LAGR slettes.

)VARS Skriver navnet på alle variable i det aktive arbeidsområdet.

)FNS Skriver navnet på alle funksjoner (program) i det aktive arbeidsområdet.

)OFF Avslutning av APL. Det aktive arbeidsområdet slettes.

)CONTINUE Avslutning av APL. Arbeidsområdet lagres automatisk ved navn CONTINUE. Dette elementet tas automatisk inn neste gang APL kalles. Elementet må slettes av brukeren med)DROP CONTINUE når han ikke trenger det lenger. Ved avslutning med @FIN direkte fra APL får en også laget et CONTINUE element.

NOEN EKSEMPLER PÅ BRUK AV OPERATORENE I APL.

I APL utføres alle uttrykk fra høyre mot venstre. Dette kan brytes v.h.a. paranteser: Noen av eksemplene er også vist med digraphs.

1) Tilordning av en enkel variable A, en vektor V, en matrise M (av V) og en tekstvektor:

```
A←1.5
:V←1 5 3 2.5 7.5 0.5 6 4 2 0.4
:M←3 3 FV
:T←'APL ER NYTTIG'
:
```

Med digraphs blir uttrykkene:

```
A $$ 1,5
M $$ 3 3 $R V'
```

Innholdet i en variabel kontrolleres ved å skrive variabelnavnet:

```

M
1          5          3
2.5        7.5        0.5
6          4          2
:
```

APL forts.

2) Multiplikasjon av A og V. Merk at resultatet skrives direkte ut på terminalen når resultatet ikke tilordnes en variabel.

```

      V×A
1.5 7.5 4.5 3.75 11.25 0.75 9 6 3 0.6
      :
```

Med digraphs: A \$X V

Alle elementer i V multipliseres med A.

3) Antall elementer i V bestemmes og legges i N samtidig som innholdet i N skrives ut v.h.a. tilordning til (digraph: #) :

```

      #←N←FV
10
      :
```

Med digraphs: # \$S N \$S \$R V

4) Indeks generatoren lager en vektor med alle tall fra 1 til N når argumentet til operatoren er N:

```

      #←I←1 12
1 2 3 4 5 6 7 8 9 10 11 12
      :
```

Med digraphs: # \$S I \$S \$J 12

5) Sum av elementene i V

```

      +/V
31.9
      :
```

6) Middelfverdi av elementene i V

```

      (+/V)÷FV
3.19
      :
```

7) Sortering av V stigende rekkefølge:

a) Indekser for stigende orden

```

      :ΔV
10 6 1 9 4 3 8 2 7 5
      :
```

b) Vektoren sortert:

```

      :V[ΔV]
0.4 0.5 1 2 2.5 3 4 5 6 7.5
      :
```

Med digraphs V[\$U V]

8) Summerer alle kolonner i M

```

      #←SUM←+/[1]M
9.5 16.5 5.5
      :
```

9) Eksempel på program

Programmet beregner integralet e^x mellom A og B (i APL * X) med midtpunktformelen:

$$I = H \sum_{j=1}^N f(A + (j - \frac{1}{2})H)$$

Funksjonen har to argumenter: N og vektoren T med intervall grensene A og B.

```

▽ R←N MIDP T
[1] 'MIDTPUNKTMETODEN'
[2] H←(T[2]-T[1])÷N
[3] S←1N
[4] SF←x(T[1]+(S-0.5)×H)
[5] I←H×+/SF
[6] 'RESULTAT: ' ; I
▽
:
```

Mer kompakt:

```

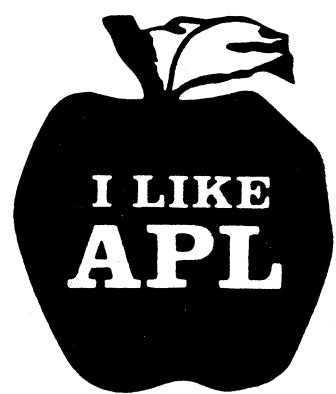
▽ R←N MP T
[1] I←H×+/×T[1]+((1N)-0.5)×H←(T[2]-T[1])÷N
[2] 'RESULTAT';I
▽
:
```

Beregning:

```

      10 MIDP 1 2
MIDTPUNKTMETODEN
RESULTAT: 4.668828682
      :

      10 MP 1 2
RESULTAT4.668828682
      :
```



MER OG BILLIGERE KJØRING PÅ 1100/62-ANLEGGET

Bruken har sommeren og høsten 1980 som vanlig bestått av mange forskjellige typer kjøring. Det har vært mer beregningskrevende kjøring enn vanlig, og det er en generell utvikling mot vesentlig mer interaktiv kjøring.

Dessuten har det skjedd en gradvis overgang fra 1100/21 til 1100/62, og i dag kjøres ca. 3/4 av jobbene på 1100/62. Kapasitetsmessig gir det ingen problemer, tvert imot er det mulig å kjøre vesentlig mer på dagtid. Dessuten er det på kvelds- og natt-tid liten belastning.

En del nøkkeltall for 1980 og 1979 er vist i en tabell nedenfor:

	okt 80	jan-okt 80
Antall satsvise kjøring	38300	240000
Antall interaktive kjøring	35400	213000
CPU-tid timer satsvis	210	1611
CPU-tid timer interaktiv	44	266
Katalogiserte filer (spor)	294000	
	okt 79	jan-okt 79
Antall satsvise kjøring	30000	262000
Antall interaktive kjøring	21000	180000
CPU-tid timer satsvis	132	1248
CPU-tid timer interaktiv	21	181
Katalogiserte filer (spor)	158000	

Det er interessant å merke seg at interaktive kjøring nå er like mange som satsvise, og at det er en betydelig vekst (26%) i CPU-uttak det siste året.

I gjennomsnitt antar vi at 1100/61 (en prosessor!) har omkring 2 ganger større sentralenhetkapasitet enn 1108 med vår jobbprofil. Dessuten kommer et tillegg p.g.a. den andre sentralenheten i vår

konfigurasjon (1100/62). Anslagsvis er økningen i forhold til 1100/61 på ca. 70%. Forbedringer i gjennomstrømningshastigheten av jobber p.g.a. "systemandel" og mindre "swapping" grunnet bl.a. et stort primærlager kommer i tillegg til ovenfornevnte forhold.

1100/62 er bygd opp av mange 4 bits mikroprosessorer (i parallell). Enkle operasjoner går hurtig, bl.a. p.g.a. et bufferlager, mens flytende operasjoner er langsomme, og må være det slik maskinen er oppbygd. Målinger har vist forholdstall for sentralenhetsforbruk fra 1:1,2 til 1:5 mellom 1100/61 og 1108 for forskjellige typer av oppgaver. For å bruke våre priser fra 1980-01-01 har vi følgelig justert opp registrert sentralenhetstid med 1,8.

Det største antall brukere kommer ut med en liten forbedring i kostnad på 1100/62 i forhold til 1108. Noen kommer vesentlig bedre ut, og noen brukere med beregningskrevende oppgaver må betale litt mer enn på 1108. Som et middel ligger det en rabatt på ca. 10% i prisen på 1100/62 i forhold til 1108. Dette skyldes kun den omtalte forskjell i forbrukt sentralenhet-tid, mens andre prispørere som inn/ut-overføringer, permanente filer, utskrift etc. er identisk med 1108's.

Som nevnt er det tunge tekniske beregninger som kommer dårlig ut i forhold til tidligere, fordi flytende tall (og dobbel presisjon) på 1100/62 ikke er raskere enn på 1108. For brukere med slike jobber skulle også NORD-500 være et tilbud. På grunn av forsinket leveranse av NORD-500, og at vi har vesentlig mer kapasitet enn tidligere, har RUNIT fra oktober 1980 foretatt prisjusteringer som gjør tidkrevende jobber mer økonomisk fordelaktig på 1100/62- maskinen.

Følgende forandringer er innført: Jobber med sentralenhet-forbruk over 7 min. pr. deloppgave får sentralenhetspris fra og med 7 min. forbruk halvert (f.eks.: 15 min. virkelig = 11 min. betalbar). Som eksempel kan nevnes at de brukere som i oktober hadde jobber med mer enn 7 minutters sentralenhet-tid pr. deloppgave (jobstep) tilsammen har fått en reduksjon i prisen på 35%.

Odd Meland

NB!
Ny pris-RINFO vil komme før jul!

SLIK FÅR DU VITE PRISEN

Ved å legge inn styrekortet '@COST' i slutten av en kjøring vil en få beregnet prisen på kjøringen. Prisberegning som blir utført av COST-prosessoren omfatter CPU-tid inn/ut-aktivitet, tilknytningstid, kostnader forbundet med leie av masselager og bruk av magnetbånd. Prioritetstillegg (fradrag) er imidlertid tatt med i beregningene. Utskrift fra COST-prosessoren ser for tiden slik ut (satsvis versjon):

"COST

OBS DISSE PRISENE ER VEILEDENDE, FAKTURA

SENTRALENHET(CPU)	KR:	.32
KANALAKTIVTET(I/O)	KR:	1.51
KORTBILDER LEST	KR:	.02
SIDER UTSKRIFT	KR:	1.06
KORTBILDER GENERERT	KR:	.00
TOTALPRIS EKS BAAND	KR:	2.91

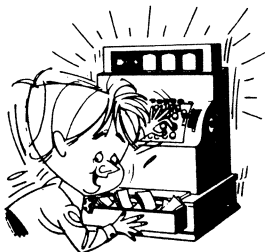
PRIS PR EVENTUELL BAANDMONTERING ER KR:

PRISENE FOR KORTLESING/STANSING OG SIDER

Et problem kan imidlertid oppstå i forbindelse med "batch"-kjøringer. Hvis et kall på COST er lagt inn i slutten på kjøringen og jobben går i feil på et tidligere stadium vil resterende styrekort (inklusive kall på @COST) vanligvis ikke bli utført. Nedenfor blir det skissert hvordan dette problemet kan løses.

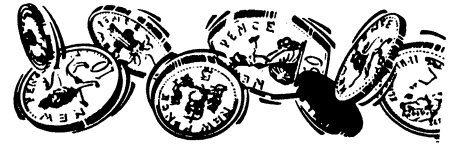
- 1) Legg inn '@SETC,I' i starten på kjøringen. Dette vil medføre at jobben ikke blir avbrutt selv om det oppstår feil i en aktivitet. Dette skulle være tilstrekkelig dersom jobben er sammensatt av uavhengige jobbtrinn slik at man ønsker å fortsette selv om en feil oppstår i et trinn.
- 2) Dersom man ikke ønsker å gå videre med etterfølgende aktiviteter dersom en feil oppstår, men likevel ønsker å få utført kall på COST i slutten av kjøringen, kan følgende oppsett benyttes:

```
@RUN...
@SYM...
@PASSWD...
@SETC,I
@jobbtrinn nr1
@TEST TE/0/S2
@JUMP PRIS
@jobbtrinn nr2
```



BRUK AV COST

```
.
.
@PRIS:
@COST
@FIN
```



Setningen '@TEST': trenger en nærmere forklaring. Det er en test på innholdet i 2. sjettedel av et tilstandsord. Her vil en finne en verdi som forteller hvordan kjøringen har forløpt så langt. I eksemplet ovenfor testes det om innholdet er 0 (dvs. ingen feilutgang har forekommet). Hvis det er tilfelle vil neste styrekort (@JUMP) hoppes over og utføring av jobben fortsetter som normalt. Hvis testen er negativ vil setningen '@JUMP..' bli utført og senere jobbtrinn hoppes over.

Nedenfor gjengis de ulike verdier vi kan finne i tilstandsordets 2. sjettedel og også en kortfattet beskrivelse av hva de ulike verdier betyr:

- 0 - Alle aktiviteter hittil i kjøringen er avsluttet på normal måte (via EXIT\$).
- 1 - Normal avslutning av siste aktivitet, feilutgang (via ERR\$) i minst en av de forutgående aktiviteter.
- 2 - Feilutgang (via ERR\$) i siste aktivitet, normal avslutning av alle forutgående aktiviteter.
- 3 - Feilutgang i siste aktivitet og en eller flere av forutgående aktiviteter.
- 4 - Alvorlig feilutgang (via ABORT\$) i siste aktivitet, normal avslutning av alle forutgående.
- 5 - Alvorlig feilutgang i siste aktivitet og i en eller flere av de forutgående.

I en og samme testsetning kan en foreta flere tester. F.eks.:

```
@TEST TE/4/T1, TE/5/S2
```

Hvis minst en av testene er oppfylt, vil en hoppe over den styresetningen som følger umiddelbart etter TEST-setningen.

For mer detaljert informasjon henvises det til UNIVAC-håndbok UP 4144.23 (PRM).

Paul Rusten

Multibanking med ASCII-FORTRAN

på 1100/62

MULTIBANKING er en teknikk som går ut på å dele et program opp i enheter kalt banker. Flere banker kan dele de samme MAP-adressene, uten nødvendigvis å dele de samme fysiske adressene. Dette er et alternativ til segmentering når man skriver et stort program. Det har den fordel at programmet kan utvide seg dynamisk i takt med hurtiglagerplassen. Blir det trangt om plassen, vil operativsystemet laste ut banker som ikke er i bruk, i stedet for hele programmer.

I ASCII FORTRAN fins muligheter for å bruke multibanking. En kan lage instruksjonsbanker som består av subrutiner, og databanker som består av commonblokker. Til dette trengs BANK- og COMPILER-setningene i FORTRAN programmet.

I tillegg må det være en kontrollbank som inneholder FORTRAN-biblioteket og alle lokale data.

Eksempel på program som har to instruksjonsbanker og to databanker:

```
@FTN,SIO ,HOVED
      COMPILER(LINK=IBJ$)
      BANK /DB1/ CB1
      BANK /DB2/ CB2
      COMMON /CB1/ A /CB2/ B
      A=1
      B=2
      PRINT*,A,B
      CALL SUB
      END
@FTN,SIO ,MAIN
      COMPILER(LINK=IBJ$)
      SUBROUTINE SUB
      PRINT*,'I SUB '
      RETURN
      END
```

Følgende MAP må brukes:



'Once Upon a Time There Was a Little System Born of Grateful and Kindly Analysts'

```
@MAP,SI PROG
LIB FTN*LIB
IBANK,MDR IB1
  IN HOVED
IBANK,DR IB2,IB1
  IN SUB
DBANK,MC DBANK
  IN HOVED,SUB1
DBANK,UD DB1 (DBANK+010000)
  IN(DBANK)F2ACTIV$( $1)
  IN CB1
DBANK,UD DB2,DB1
  IN(DBANK)F2ACTIV$( $3)
  IN CB2
END
```

. Forklaringer

- . Instruksjonsbanken IB1:
- . HOVED's instruksjoner
- . Instruksjonsbanken IB2:
- . SUB's instruksjoner
- . Kontrollbanken:
- . lokale data (+ FTN*LIB)
- . Databanken DB1:
- . element fra FTN*LIB
- . commonblokken CB1
- . Databanken DB2
- . element fra FTN*LIB
- . commonblokken CB2

Map-kart:

Et program kan bestå av inntil 250 banker. Teknikken kan dessverre ikke brukes sammen med noen av de nåværende programbibliotekene ved RUNIT, men det vil med tid og stunder komme et "FTN multibank NAG library". Dette kan brukes i multibankede programmer, og inneholder bl.a. rutiner for å dele opp matriser og legge delmatrisene i databanker for å spare plass.

Det er laget et mer omfattende notat om multibanking som kan fås hos RUNIT's Brukerkontaktgruppe, tlf. 3029.



'Waiter, There's a Diode in My Soup.'

Fridtjov Iversen

JOBBER SOM HENGER



VRANGLÅS - PROBLEMET

En vranglås er en tilstand hvor to eller flere jobber gjensidig hindrer hverandre i å komme videre. Når en vranglås oppstår, kommer dette vanligvis av at flere jobber forsøker å tilordne de samme filene samtidig. Ved betjent kjøring vil operatøren oppdage forholdet og løse floken ved å terminere en eller flere jobber. Ved ubetjent kjøring vil vranglåsen forbli uløst. Da systemet har en øvre grense for antall åpne satsvise jobber (vanligvis 6), vil utførelsen av satsvise jobber stanse helt dersom kvoten fylles av jobber i vranglås.

I det følgende skal vises hvordan vranglås kan oppstå, og hvordan dette kan forhindres ved en hensiktsmessig utforming av kjøreoppsettet. Fremstillingen er ikke uttømmende, spesielt interesserte henvises til faglitteraturen.

OS1100 (operativsystemet for Univac's 1100-serie) tilbyr to grader av filtilordning.

Tilordning med X-opsjon gir eksklusiv tilgang til filen. Denne formen for tilordning brukes alltid når det skal skrives på filen, slik at samtidig lesing eller skriving fra andre jobber forhindres. Tilordning med X-opsjon er betinget av at ingen andre jobber har filen tilordnet.

Tilordning uten X-opsjon greier seg dersom det bare skal leses fra filen. Flere jobber kan ha filen tilordnet uten X-opsjon samtidig. Tilordningen er betinget av at ingen andre jobber har filen tilordnet med X-opsjon.

Ved begge former for tilordning stiller systemet visse betingelser som må være oppfylt av de jobbene som tilfeldigvis kjører samtidig. Dersom betingelsene ikke er oppfylt, må jobben vente. Dersom to eller flere jobber gjensidig venter på hverandre, oppstår en vranglås.

Det kan være vanskelig å forutse mulighetene for vranglås. Filer kan nemlig tilordnes ikke bare via @ASG-setninger i kjøreoppsettet, men også direkte fra et program. Kompilatorer, editor, FURPUR etc. vil automatisk tilordne filene som angis på styrekortet. De filene som det skal skrives på blir tilordnet eksklusivt. Når prosessoren terminerer, fraordnes filene igjen.

Eksempel:

```
Jobb A:  @RUN ....
         @ASG,A PROG.
         @FOR,IS PROG.A
         .
         .
         @FIN
```

```
Jobb B:  @RUN ....
         @ASG,A PROG.
         @FOR,IS PROG.B
         .
         .
         @FIN
```

Dersom disse jobbene kjøres samtidig, vil vranglås oppstå. I begge jobbene vil Fortran-kompilatoren utføre en @ASG,AX PROG. Da filen for begge jobbers vedkommende er tilordnet en annen jobb, må jobben vente.

Vranglås forhindres dersom en av følgende retningslinjer følges for utforming av kjøreoppsettet:

1) Dersom alle referanser til filer i jobben skjer via Univac's prosessorer (kompilatorer, editor, FURPUR, ELT, DATA osv.) overlates all tilordning og fraordning til disse. Jobben inneholder da ingen @ASG-setninger.

eller:

2) Alle filer som benyttes i kjøringen tilordnes med X-opsjon umiddelbart etter @RUN. Jobben vil da ikke bli åpnet før alle filene er tilordnet.

Har du glemt brukernummeret eller passordet ditt?

Ta kontakt med RUNIT's ekspedisjon i 2. etg. SB2, evt. terminalen på Rosenborg, Lade eller MTS, som vil hjelpe deg å finne tak i brukernummer/passord. Husk å ta med legitimasjon (for studenter, studiebevis).

Altså, ta ikke ut nytt brukernummer om du har glemt brukernummeret eller passordet ditt!

RUNIT's driftsgruppe



INDEX — TIDLIGERE

ARTIKLER I RUN-NYTT

Diverse

- | | | | |
|--|--|--|--------------------------------|
| Brukerkontakten styrkes | :Nr.3-79 S. 2 | Lov om personregistre | :Nr.3-79 S. 3
:Nr.2-80 S.24 |
| Dataetikk | :Nr.8-77 S. 3
:Nr.2-80 S.24 | Mikroprosessorutstyr i
NTH - miljøet | :Nr.3-79 S. 9 |
| Dataforkortelser: | | Minimaskiner i UNIT-miljøet | :Nr.2-80 S.29 |
| DMA | :Nr.7-77 S. 9 | NORD-planer | :Nr.2-80 S.16 |
| GIGO | :Nr.2-79 S. 2 | NSD | |
| OCR | :Nr.4-78 S.15 | Norsk Samfunnsvitenskap-
elig Datatjeneste på Lade | :Nr.8-77 S.22 |
| OEM | :Nr.1-78 S.11 | NSD's Kretsdatabank | |
| Datakunst | :Nr.1-78 S.12
:Nr.2-78 S.12
:Nr.3-78 S.12
:Nr.4-78 S. 2 | Inneholder folketellings-
opplysninger fra folke-
tellingen i 1960 og 1970 | :Nr.1-78 S.22 |
| Datanett | :Nr.1-78 S. 4 | Ny UNIVAC 1100 maskin | :Nr.1-80 S.17
:Nr.2-80 S.11 |
| Datasikkerhet | :Nr.2-80 S.24 | Nytt operativsystem | :Nr.2-80 S.11 |
| Datasjakk | :Nr.2-80 S.17 | Orakeltjenesten | :Nr.2-80 S.35 |
| Dataspråk | :Nr.2-79 S. 3
:Nr.2-79 S.27
:Nr.2-80 S.27 | Passord | :Nr.3-78 S. 4 |
| Den Norske Dataforening | :Nr.2-80 S.34 | Personal Computing | |
| Dividerer du med null i
dine FORTRAN-programmer | :Nr.1-79 S. 7 | Nytt databegrep | :Nr.3-78 S. 2 |
| DMU- Datamaskinutvalget
ved Universitetet i Tr.heim | :Nr.2-80 S.22 | I NTH-miljøet | :Nr.3-79 S. 4 |
| Elektronikk som kan snakke | :Nr.1-80 S. 2 | Presentasjon av RUNITS
grupper | |
| Erfaring med bruk av mikro-
datamaskiner | :Nr.2-79 S.12 | Gruppe for informasjons-
systemer | :Nr.8-77 S.15 |
| Håndbøker | | Teknisk gruppe | :Nr.2-78 S.20
:Nr.3-78 S.13 |
| fra RUNIT | :Nr.2-80 S.13 | Priser ved RUNIT fra 1980-
01-01 | :Nr.1-79 S.14 |
| fra UNIVAC | :Nr.3-78 S. 6
:Nr.1-80 S.22 | Privatdatamaskin benyttet
som intelligent terminal | :Nr.3-79 S.11 |
| Innlesing av papirhullbånd | :Nr.2-78 S. 7 | På besøk i lokale datamiljø: | |
| Kjøreplan | :Nr.1-80 S.13 | Kolbjørn Bell | :Nr.1-80 S. 4 |
| Kompjuter- Datastudentenes
linjeforening | :Nr.2-80 S.40 | SINTEF avd. for konstruk-
sjonsteknikk | :Nr.2-80 S.10 |
| | | Humanistene på Lade/Dragvoll | :Nr.3-78 S.16 |
| | | Reguleringsteknikk | :Nr.8-77 S.16 |
| | | Universitetet på Lade | :Nr.2-79 S.12 |
| | | Vassdrags- og Havnrelab. | :Nr.2-78 S.14 |

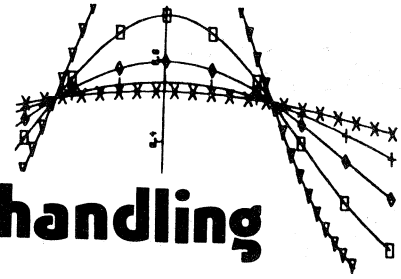
INDEX forts.

Rammeavtaler	:Nr.2-79 S. 4
RUNIT med tilbud på små-maskinsiden	:Nr.8-77 S.20
Stansing av hullkort på 1100-21	:Nr.4-78 S.10
Statens standard avtale-formular for EDB-anskaffelser	:Nr.1-80 S.10
Statistikkmisbruk generert av datamaskin	:Nr.2-78 S.10
Teknologiske utviklingstendenser	:Nr.3-78 S. 8
Terminering ved 'MAX TIME'	:Nr.2-78 S. 6
Tidtakermaskinen Jensen	:Nr.8-77 S. 2
UNINETT	:Nr.1-78 S. 5
Tilgjengelig for brukere nå	:Nr.3-79 S.13
RUNIT søker etter brukere til UNINETT	:Nr.3-79 S.14
Nettet tilbyr	:Nr.3-79 S.15
Har du noe å tilby nettet	:Nr.3-79 S.15
"Lure" program laget av datastudenter	:Nr.1-80 S.23
Utstansing av hullkort	:Nr.1-80 S.22
Uttesting av programvare for mikroprosessor	:Nr.2-78 S.11
Utstyr og tjenester	:Nr.3-79 S.19

Filer og filbruk

Filen som ble vekk	:Nr.1-79 S. 2
	:Nr.1-78 S. 2
Diverse forslag til hvorfor og hvordan rette opp	:Nr.2-78 S. 2
	:Nr.4-78 S. 8
Gjennvinning av slettede elementer	:Nr.2-80 S. 7
Kontroll av katalogiserte filer	:Nr.2-78 S. 8
Pakk filen - spar penger	:Nr.2-79 S. 5
Sikring av filer	:Nr.2-78 S. 5
	:Nr.2-80 S.25

Grafisk databehandling



Bedre grafisk utstyrs-tilbud	:Nr.7-77 S. 4
Grafiske enheter tilknyttet Nord-10 som igjen er tilknyttet Univac 1100/21 over kommunikasjonslinje.	:Nr.8-77 S. 9
CALCOMP 936 plotter	:Nr.1-78 S.20
GPGS-F	
Endringer i GPGS-F USERS GUIDE	:Nr.1-78 S.20
Ny versjon	:Nr.2-79 S.20
Driver for Tektronix 4662 plotter	:Nr.8-77 S. 9
Grafisk databehandling	
Oversikt over utstyr og hva dette kan brukes til	:Nr.2-78 S.17
Plotting av kurver, histogrammer m.m.	:Nr.2-78 S.17
SURRENDER	
Presentasjon av 3-dimensjonale data	:Nr.2-79 S.18
TEGRUT	:Nr.2-78 S.18
Tektronix digitaliseringsbord	:Nr.2-80 S.16

Magnetbånd

Bruk av private magnetbånd	:Nr.8-77 S.11
Endret typebetegnelse for 9-spor 800 bpi magnetbånd	:Nr.1-80 s.22
Magnetbånd til og fra andre anlegg	:Nr.1-80 S.21
Magnetbåndutstyr ved RUNIT	:Nr.3-78 S.11
Nye priser på magnetbånd	:Nr.1-80 S.21
Plass på magnetbånd	:Nr.3-78 S.11
Sikring av filer og magnetbånd på UNIVAC	:Nr.1-80 S.20
Til brukere av magnetbånd på UNIVAC	:Nr.1-80 S.20

INDEX forts.**Programvare**

Digital signalbehandling	:Nr.2-80 S.15
EISPACK	
En samling FORTRAN sub- rutiner som løser standard egenproblem for diverse typer matriser	:Nr.2-79 S. 9
GLIM	
Algoritme for tilpassing av generaliserte lineære modeller	:Nr.2-80 S.14
NAG,RUNITS numeriske programbibliotek	:Nr.3-78 S.13 :Nr.4-78 S.16 :Nr.2-79 S. 8 :Nr.1-80 S.18
NO308 GAUSSN	
En FORTRAN subrutine som beregner et n-dimensjonalt intergral	:Nr.2-79 S. 9
OPTIMA 1100	
Programsystem for plan- legging og oppfølging av nettverksorienterte prosjekter	:Nr.1-80 S.18
RKF 45	
Program for numerisk løs- ning av ordinære differen- sialligninger	:Nr.3-78 S. 7
SPSS	
Spørsmål, råd og vink	:Nr.3-78 S.10
Spørsmål, råd og vink	:Nr.4-78 S.18
Nye SPSS-versjoner	:Nr.1-80 S.19
SPSS-informasjon	:Nr.1-80 S.19
STATISTIKKPROGRAMMER	
BMDP	:Nr.2-80 S.14
COFAMM	:Nr.2-80 S.15
EFAP	:Nr.2-80 S.15
EXPAK	:Nr.2-80 S.15
LISREL IV	:Nr.2-80 S.15
MULTIVARIANCE	:Nr.2-80 S.16
En brukers erfaring	:Nr.2-80 S.16
PSTAT	:Nr.2-80 S.14
UKILT-1100	
Programmer for løsning av nettverksproblemer	:Nr.2-79 S. 9

Prosessorer

Bruk av @PRT	:Nr.2-78 S. 8
DOC-prosessor	:Nr.3-78 S.10
EDITOR	
Effektiv sletting.	
Bruk av D+	:Nr.1-78 S.18
Bedre sikring mot tap av oppdaterte elemente	:Nr.7-77 S. 5
NEWS	
Proseszor som gir utskrift av aktuelle nyheter	:Nr.1-78 S.16
STATUS	
Informasjon om jobber under utførelse i maskinen	:N.7-77 S. 8 :Nr.2-78 S. 6
Styring av utskrift-/ utskriftsenheter	:Nr.1-78 S.15
SUSPEND og RESUME	
Hjelpemiddel for styring av utskrift ved inter- aktiv kjøring	:Nr.8-77 S.10
Tekstredigering med DOC	:Nr.8-77 S. 5

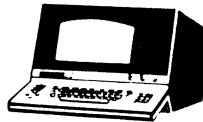
Språk

APL (A programming Language)	:Nr.8-77 S.12 :Nr.3-78 S.15 :Nr.2-80 S.12
ASCII-COBOL	:Nr.3-78 S. 9
Level 4R1 (ny versjon)	:Nr.4-78 S.11
ASCII-FORTRAN	:Nr.2-80 S. 4
En brukers erfaring	:Nr.4-78 S. 6
CTS	
Effektiviser program- utviklingen med CTS	:Nr.3-79 S.16
CTS gjør databehand- ling lettere	:Nr.2-80 S.18
Brukervennlig dialog med CTS	:Nr.2-80 S.21

INDEX forts.

FORTTRAN-oversettere FORTTRAN V, ASCII-FORTTRAN, Reentrant FORTTRAN	:Nr.4-78 S. 4
Flyttbare FORTTRAN- programmer	:Nr.2-79 S. 6 :Nr.1-80 S. 5
INDIRA Interaktivt Databasespråk for Ikke-programmerere mot RA2	:Nr.1-78 S.10
MARY Et masinorientert programmeringsspråk	:Nr.4-78 S.12
SIMULA Versjon 2.OMZ av Simula-oversetteren Ny SIMULA-kompilator	:Nr.2-78 S. 2 :Nr.1-79 S.10

ASCII-terminaler	:Nr.4-78 S.10
Bruk @BRKPT og @SYM riktig!	:Nr.2-79 S.10
Enkel tilknytning av avanserte terminaler	:Nr.4-78 S.10
Leie av bordterminal Kan tilknyttes UNITS hovedanlegg på oppringt linje	:Nr.7-77 S. 4
Nye Tandberg-terminaler	:Nr.2-80 S.28
Oppringt samband	:Nr.2-79 S.10
Programvare fra University of Maryland Program som er beregnet på avansert terminalbruk	:Nr.8-77 S.14
Skikk og bruk for terminalsittere	:Nr.1-78 S. 7
Startjobber fra terminal	:Nr.2-79 S.10
Studentkjøring via terminal	:Nr.1-78 S.14
Styring av utskrift til interaktiv terminal	:Nr.2-79 S.10
Terminal-kokebok	:Nr.1-78 S. 7



Terminaler — bruk og utstyr

Slik får du tilknyttet terminal

Det er mulig å få terminaltilknytning til følgende utstyr ved RUNIT

Univac 1100/21
Univac 1100/60
Nord 100

Terminaler som skal kommunisere med data-maskiner ved RUNIT, må enten tilkoples via oppringt samband eller via fast oppkoblede linjer.

OPPRINGT SAMBAND

Interaktive terminaler mot Univac 1100/21 og Univac 1100/60 via terminalkonsentratoren PDP 11/20:

- Tlf. 40703, 93037, 93050
- TTY kompatibel terminal
- Overføringshastighet 300 biter/s
- Asynkron overføring
- Full dupleks

Interaktiv terminal mot Univac 1100/60

- Tlf.40706
- Overføringshastighet 2400 biter/s
- Uniscope 100/200
- Synkron overføring
- Halv dupleks
- Identifikator S13U01

Blokkterminal mot Univac 1100/21

- Tlf.40700
- NTR (Nine Thousand Remote)
- Overføringshastighet 2400 biter/s
- Synkron overføring
- Halv dupleks
- Identifikator (etter avtale)

Interaktiv terminal mot Nord-100

- Tlf.93039
- TTY kompatibel terminal
- Overføringshastighet 300 biter/s
- Asynkron overføring
- Full dupleks

Det er planer om ytterligere utvidelser av innganger. I første omgang TTY-inngang med overføringshastighet 1200 biter/s. Dessuten vil det før årsskiftet 1980/81 bli inngang mot Univac 1100/60 for UTS-400 terminaler. Den oppringte blokkterminalinngangen på tlf. 40700 vil utgå på grunn av minimal bruk.

Slik får du forts.

FAST OPPKOPLEDE LINJER

Bestilling av terminalinnganger sendes skriftlig til undertegnede.

Bestilling av linjer sendes til Televerket via NTH's driftsavdeling v/ overing. Langøy. Der hvor det finnes private kabler, bestilles linjer hos teknisk gruppe, RUNIT.

Terminaler som tilkoples via Televerkets linjenett, må benytte modem. Modem bestilles normalt direkte fra Televerket. Terminaler plassert lokalt på Gløshaugen, kan imidlertid benytte en modemtype som bestilles via undertegnede.

Forespørsler om terminaltilkopling kan rettes til RUNITs driftsgruppe v/ Torgnes, tlf. (9)3021 evt. (9)2952.

Snorre Torgnes

Angående terminal-konsentrator PDP11A og PDP11B

1) @@-kommandoer kan nå skrives i "små" og "store" bokstaver.

2) Lovlige @@-kommandoer:

2.1 @@TERM

Terminerer linjen til U1100.

2.2 @@PTI

Setter PDP-11 og U1100 i papirbånd input tilstand.

2.3 @@END

Forårsaker at PDP-11 og U1100 går ut av papirbånd input tilstand, samt terminerer andre '@@'-kommandoer.

2.4 @@TYPE <TERMINALTYPE> eller

@@TYPE=N

Omkonfigurerer en terminalinngang til ønsket terminaltype.

2.5 Lovlige terminaltyper

TYPE	N
TTY	0
SILENT	1
TEKTRONIX 4010	2
ALPHA	3
TTY 38	4
TEKTRONIX 4014	5
DECWRITER	6
DCT 500	7

Eks.: @@TYPE ALPHA

@@TYPE=3

Betyr det samme, og konfigurerer inngangen om til å gjelde skjermterminal.

3. @@BPS <parameter> eller

@@BPS

Omkonfigurerer en terminalinngang til ny hastighet.

NB! En bruker kan ikke sette hastigheten høyere enn det som terminalinngangen opprinnelig er konfigurert til.

3.1 Lovlige parametre til @@BPS-KOMMANDO:

110
300
600
1200
2400
4800

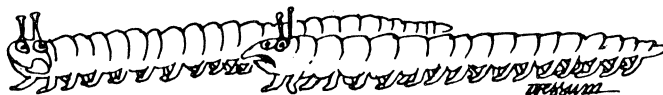
Eks.: @@BPS 1200

Vil sette ny hastighet = 1200 BPS på terminalinngangen.

@@BPS

(Uten parameter) vil resette terminalinngangen til den hastighet den normalt er konfigurert til.

Per Arne Enstad



"No, it's 15 pair, I'm sure of it. I walked over a pocket calculator."

På besøk i de lokale datamiljøer: Neurologisk avdeling, Regionsykehuset

Det er minst 3 avdelinger ved Regionsykehuset hvor datateknikken har gjort sitt inntog. Ved Høresentralen har de tatt i bruk en Rockwell Aim 65 til automatisering av hørselsprøver, og et MYCRON-anlegg brukes bl.a. til automatisk registrering av og statistisk analyse av hjertedata og utregning av hjertevolum ved Klinisk-fysiokjemisk laboratorium.

RUN-NYTT har besøkt Neurologisk Avdeling, der de bruker Apple privatdatamaskin for å analysere data for muskelstyrke hos Multipel Sklerose-pasienter.

For vel et år siden ble medisinerstudent Vidar Berg en hyppig gjest hos RUNIT, og det var privatdatamaskiner han forhørte seg om. Brått ble han borte, og det viste seg at han hadde kjøpt seg egen Apple-maskin, tatt permisjon fra studiet og blitt stipendiat ved Neurologisk laboratorium.

Berg har alltid hatt interesse av elektronikk, og da professor Sjaastad hadde flere interessante prosjekter innen neurologi på gang, prosjekter med sterkt elektronisk innslag, var det lett å lokke han med.

Ved neurologisk lab driver de bl.a. med epidural ryggmargstimulering av Multipel Sklerose (MS) - pasienter. Det vil si at to elektroder settes inn i ryggstølen, og epiduralt vil si på utsiden av ryggmargen. En radiomottager på utsiden av ryggen er forbundet med elektrodene, og ved hjelp av

en liten bærbar radiosender kan det overføres strøm til elektrodene. Denne strømstimuleringen av ryggmargsnervene styres av pasienten selv, og hensikten er å styrke og kontrollere muskelbevegelsene.

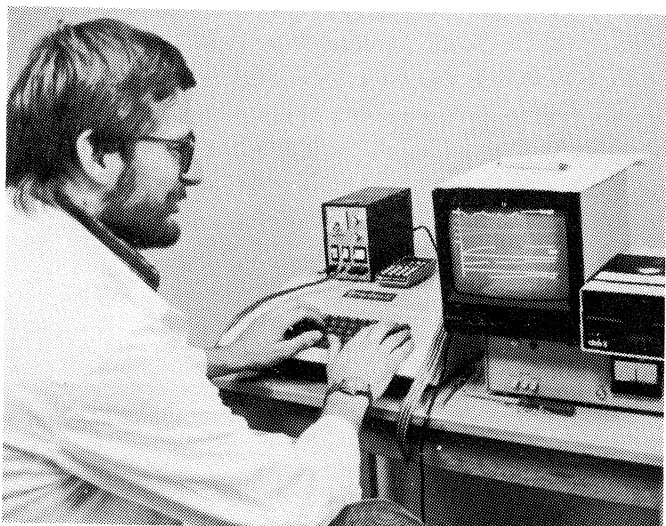
MS-pasienter har problemer både med spasmer - ukontrollerte rykninger i kroppen - og ufrivillig vannlating. Ved hjelp av den elektriske stimuleringen bringes musklene bedre under kontroll.

For å måle muskelkraft og muskelbevegelse benyttes et spesielt apparat, og det er et program på Apple for å analysere disse målingene, som Berg har vært med på å utvikle. Han begynte prosjektet på sin private Apple, men neurologisk lab fant ut at dette verktøyet var så nyttig at lab'en anskaffet sin egen maskin.

Berg har nå begynt på sitt siste studieår, og i samarbeide med sivilingeniør Håkon Høye arbeides det med videre utbygging av dataprogrammene. Høye har laget et program som registrerer og analyserer tiden mellom hvert hjerteslag, dette har betydning ved spesielle sykdomsbilder. Andre planer de har ved avdelingen er automatisk registrering av pupillestørrelse og diagnostisering i den forbindelse. Maskinen ble egentlig innkjøpt for et prosjekt som ennå ikke har startet, måling og analyse av hjerteaktivitet. I denne forbindelse vil maskinen bli brukt til simulering av teorier bak dette.

For undersøkelse av MS-pasienter er det også planer om videreutvikling, med bl.a. direkte registrering av muskelmålingene i maskinen ved hjelp av analog /digital omformer. Dermed får man fram sammenligninger mellom elektrisk muskelaktivitet og muskelkraft. Dette prosjektet har nå fått støtte fra NAVF.

I forbindelse med sin private bruk av Apple, kan Berg fortelle at den spillelidenskapen som vanligvis blusser opp ved slike anskaffelser, for hans tilfelle ble kortvarig. Han har nå gått over på mer seriøse anvendelser som f.eks. bilregnskapsprogram. På lengre sikt planlegger han både mer primærlager, PASCAL og utskriftsenhet, og i forbindelse med at han går over i aktiv legetjeneste har han tenkt å utvikle program for legeregnskap og pasientarkiv.



Her ser vi Vidar Berg utprøver program for direkte registrering av muskelaktivitet.

RUNIT'S BIBLIOTEK



RUNIT og IDB har et felles fagbibliotek, som drives av RUNIT, i 6. etg. Sentralbygg II.

Biblioteket er godt benyttet, både av studenter, RUNIT- og IDB-ansatte og andre fra NTH/Sintef-miljøet.

Som en kan se av bildene er biblioteket utstyrt med to arbeidsplasser og en "lese krok" med tidsskrifthyller. Eldre nummer av tidsskriftene er til utlån. Det abonneres på ca. 100 tidsskrifter, de mest kjente fagtidsskriftene og en del ikke-faglige. Av de faglige tidsskriftene kan vi nevne Datamation, IBM Systems Journal, Software Practice and Experience, samt de aller fleste av tidsskriftene som utgis av Association for Computing Machinery.

Biblioteket har også et rikholdig utvalg av tidsskrifter for de små datasystemene ("personal computing"). Vi nevner Byte, Creative Computing, Personal Computing, Interface Age og Microprocessors and Microsystems.

Det finnes ca. 5000 bøker og rapporter, både faglige og ikkefaglige, og nytt av året er også pensumsbøker for IDB og Elektro (ikke til utlån).

Ellers er det anledning til å sette seg ned for å lese Adresseavisen, Aftenposten, Donald Duck (ca. 10 kassetter) eller studere kart, turist-guider, flyruter, rutebok for Norge m.m.



Teknisk/matematisk gruppe ved RUNIT består av 15 personer og sitter i 5. og 6. etg. i Sentralbygg II.

Gruppens aktiviteter er delt i to:

- Visuell databehandling og DAK (DatamaskinAssistert Konstruksjon)
- Teknisk databehandling

De viktigste prosjektene innen teknisk databehandling er:

- E/P-data (Analyse av Nordsjødata). Dette er det største prosjektet og utgjør ca. 2 1/2 årsverk pr. år
- Konvertering av SPSS til Nord-500
- Diverse oppdrag for Direktoratet for vilt og ferskvannsfisk

Prosjekter innen visuell databehandling og DAK:

- Grafisk nettverk
- Oppbygging av mann/maskin interaksjon laboratorium
- Videreutvikling, salg og vedlikehold av GPGS-F
- SPICS-prosjektet - standardisert programbibliotek i interaktive CAD (Computer Aided Design)-systemer
- Tysk/norsk samarbeid om DAK/DAP Sammen med Sintef avd. 17 og 18
- Konsulentvirksomhet på grafisk databehandling og DAK
- I oppstarting: IDIGS-norsk forslag til internasjonal standard (videreføring av GPGS-F, NTNf-støttet)

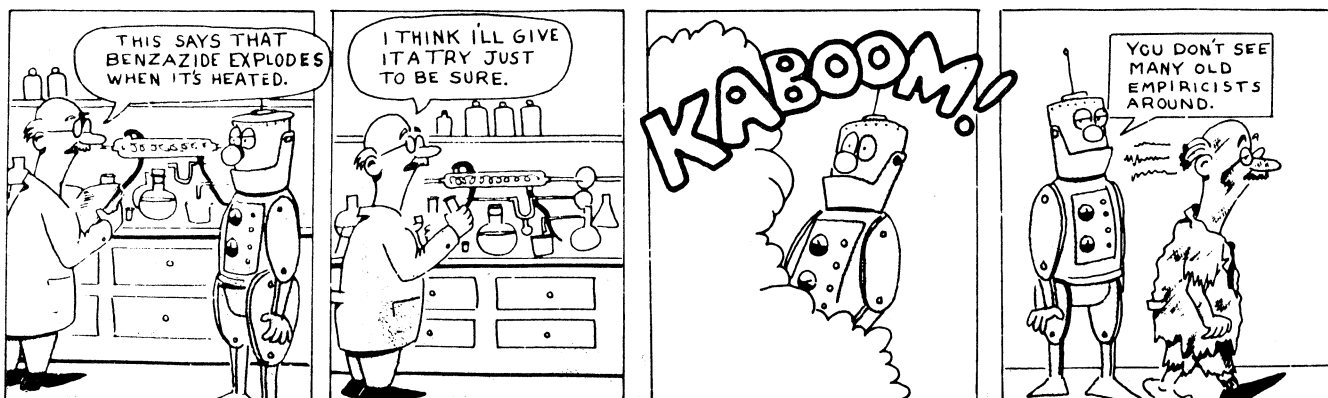
Gruppen har vært meget aktiv på området grafisk databehandling og har markert seg både nasjonalt og internasjonalt.



GEIR MOE

De har også en del kursvirksomhet i forbindelse med GPGS-F, foreleser i grafisk databehandling ved Institutt for databehandling og Trondheim Ingeniørhøgskole, og har gitt en rekke forelesninger på internasjonale kurser, seminarer etc. Tre av gruppe-medlemmene har dessuten vært formenn i NORSIGD (Norsk samarbeid innen grafisk databehandling).

Geir Moe overtok som gruppeleder i juni etter Ketil Bø som fungerte i 8 år.



TEST DITT HODE DU RASKE GUTT

Julen nærmer seg, og derfor har vi anrettet et fat med julenøtter: problemer av varierende vanskelighetsgrad som skal løses. Problemer har vel de fleste nok av fra før, men disse er ment å være et hyggelig tidsfordriv. Viser de seg å være for lite hyggelige, er det bare å glemme dem!

Hunkjønnet må også gjerne prøve seg, når overskriften adresserer seg til gutter, er det fordi det passet så godt med sangtittelen vi har plagiert.

Svarene står annet sted i bladet. Lykke til!



KRYSSSTALL

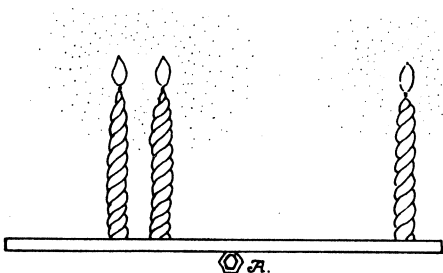
For en datamaskin som var programmert til å løse oppgaven nedenfor, ville den være unnagjort på 0.014 sekunder. En menneskelig hjerne vil nok bruke noe lengre tid, men oppgaven lar seg i alle fall løse. Tallene 1 til 9 skal plasseres i de ledige rutene slik at regnestykkene stemmer. Hvert tall skal bare brukes en gang, og noen er allerede satt på plass.

7	+		+		=3
+		+		-	
	+		x	3	=9
-		x		+	
	-		+		=2
=4	=4	=6			

	x		-		=4
-		+		+	
	+	8	+		=3
+		-		x	
	+		+		=4
=8	=4	=8			

LYSJUSTERING

Hvis du ikke klarer å resonnerer deg fram til løsningen på oppgaven nedenfor, kan du forsøke å løse den praktisk i løpet av julen. De tre julelysene er plasert slik at vektstangen akkurat balanserer, men hva vil skje når de har brent en stund - vil det fortsatt være balanse?



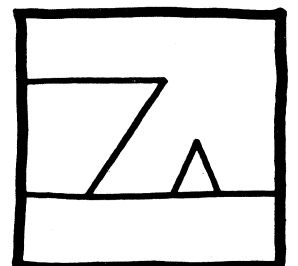
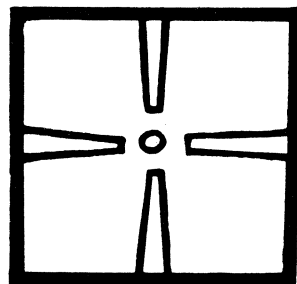
EGNER DU DEG SOM SAMLEBÅNDSARBEIDER?

Å stå ved et samleband og studere de produkter som seiler forbi for å sortere ut de med feil, vil vel for de fleste fortone seg som kjedelig arbeid. Dette har RUNIT tenkt å gjøre noe med. I samarbeid med Raufoss ammunisjonsfabrikk er et prosjekt igang for utsortering av defekte patronhylser. Grafisk databehandling skal gjøre jobben uten menneskelig innblanding.

For at RUN-NYTTs lesere skal få en føling med hvor stor tjeneste RUNIT gjør mennesket ved dette prosjektet, har vi laget en prøve på lesernes observasjonsevne. Alt fra ingen til flere eksemplarer kan være defekte på samlebandene nedenfor. Med utgangspunkt i modellen til venstre skal feilvarene plukkes ut i løpet av 4 minutter eller mindre.

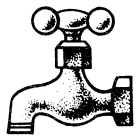
DRODLER

Drodler var med forrige gang vi hadde julenøtter-spalte i RUN-NYTT, og vi kan ikke være dårligere denne gang heller, Spørsmålet er: Hva forestiller tegningene nedenfor?

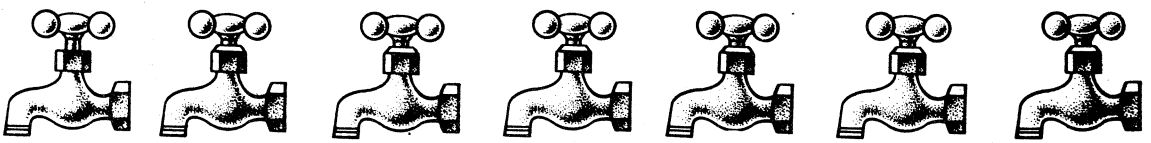




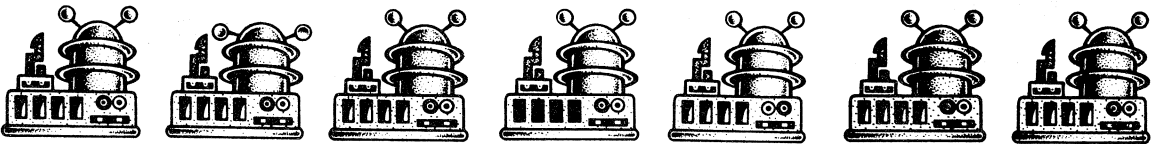
1.



2.



3.



4.



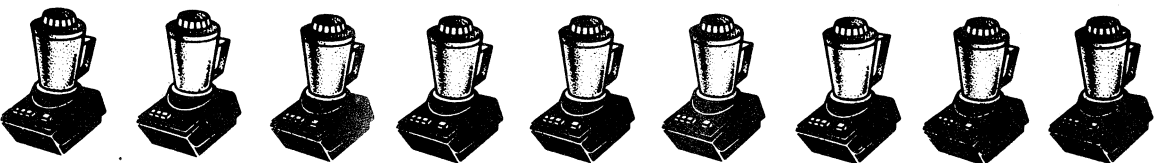
5.



6.



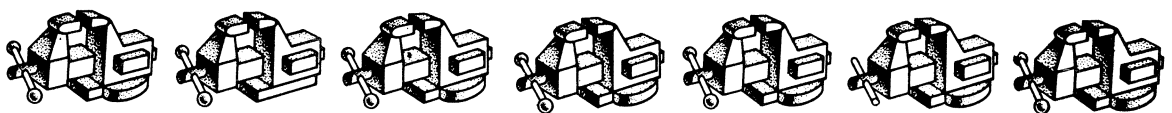
7.



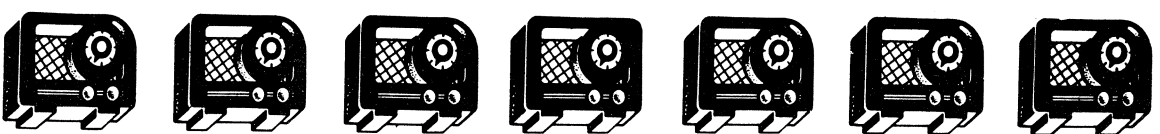
8.



9.



10.



11.

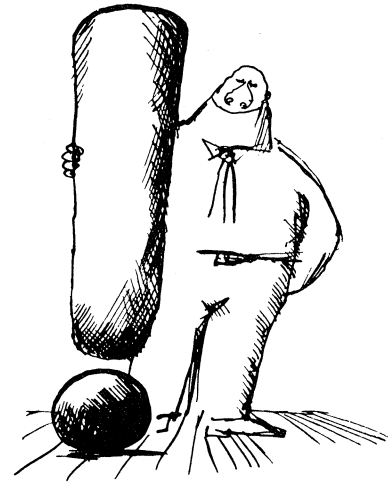


SVAR, JULENÖTTER

KRYSSSTALL

7	+	8	+	5	=3
+		+		-	
6	+	2	x	3	=9
-		x		+	
9	-	1	+	4	=2
=4	=4	=6			

2	x	5	-	6	=4
-		+		+	
1	+	8	+	3	=3
+		-		x	
7	+	9	+	4	=4
=8	=4	=8			



SAMLEBÅND

Følgende hadde defekter:

1. nr. 4 og 6
2. nr. 1
3. nr. 2,4 og 5
4. ingen
5. nr. 3
6. nr. 5
7. nr. 2 og 7
8. ingen
9. nr. 2 og 6
10. nr. 4
11. nr. 1,2 og 7

LYSJUSTERING

Hvis lysene brenner nøyaktig like fort, vil vekten holde seg i balanse. Det er loven om kraft gange arm som kommer inn, og selv om lysene mister halvparten av sin vekt, vil dette produktet være det samme på begge sider av balansepunktet. En praktisk prøve vil vise at lysene ikke brenner like fort, og at det derfor vil oppstå ubalanse likevel.

DRODLER

Den ene viser fire elefanter som snuser på en grapefrukt, på den andre er det et skip som kommer akkurat for sent til å redde en druknende heks.

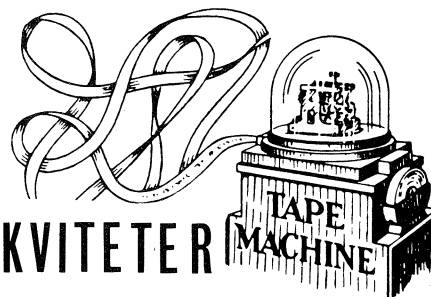
FORSIDEN

Spiller nr.1 ser på feil side, nr.2 og 3 har oppmerksomheten rettet om riktig side.

BLØTKAKEKALAS

Ved VHL har de nettopp laget en utmerket programkatalog, og brukerkontaktgruppa bisto med å få den trykket ut ved hjelp av DOC-prosessoren. Den fornøyde sjef for prosjektet, Aimar Sørensen (t.v.), kvitterte for assistansen med å invitere de impliserte på bløtkake. Et eksempel til etterfølgelse!





DATA-ANTIKVITETER

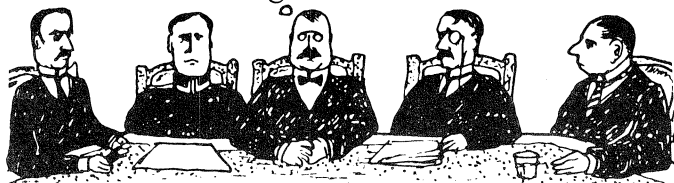
At datautstyr fort blir gammeldags, er vel et kjent fenomen for de fleste. Det er nå snart tyve år siden dataalderen begynte ved NTH, og oppsamling av antikvariske data-gjenstander har såvidt begynt. GIER, den første datamaskin i miljøet, er stedt til foreløpig hvile på datalaboratoriet til Institutt for Databehandling. Etterhvert som masselagerutstyr o.l. er kassert, er sentrale komponenter plukket av og gjemt. For 1108 som nylig gikk ut av tjeneste blir flere av de sentrale organer (hele skap) gjemt for ettertiden. Samlemanien omfatter ikke bare maskinvaren, håndbøker o.l. er også en del av datahistorien og bør i en viss utstrekning gjemmes.

Foreløpig blir det meste bare stuet bort på et lager, i håp om at når datadelen av universitetet en gang blir samlet i eget hus, kan det her opprettes et lite data-museum. Med tanke på dette er vi interessert i å få tips fra andre i universitetsmiljøet som kan ha ting av datahistorisk interesse. Gi i såfall beskjed til RUN-NYTT's redaksjon.

Som et apropos kan nevnes at det ble utlyst en liten intern konkurranse på RUNIT om hvem som hadde det eldste hullkortet. Det eldste vi fikk inn var fra den aller første UNIVAC 1107 tiden, ca. 1965. Kortet hadde vært brukt som bokmerke! På samme måte som gamle posthornsfrimerker har også hullkort gjennomgått små detaljforandringer gjennom tidene.

For spesielt interesserte bringer vi en kopi av UNITS eldste hullkort:

Hva tenker DMU-formannen?



Oppslutningen om denne konkurransen var ikke akkurat enorm. Tre bidrag kom redaksjonen i hende, fra totalt to bidragsytere:

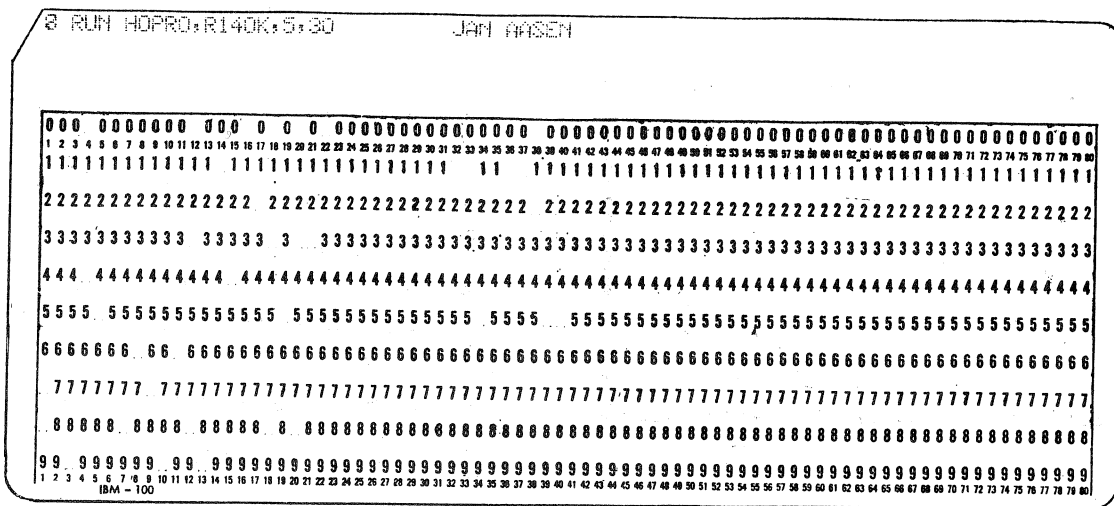
Per Ditlef Øvrebø, RUNIT:

- 1) "(Gulp) - noe så flaut - eneste deltager med sløyfe...."
- 2) "Håper tegneren blir ferdig snart, nå har vi sittet slik i to timer."

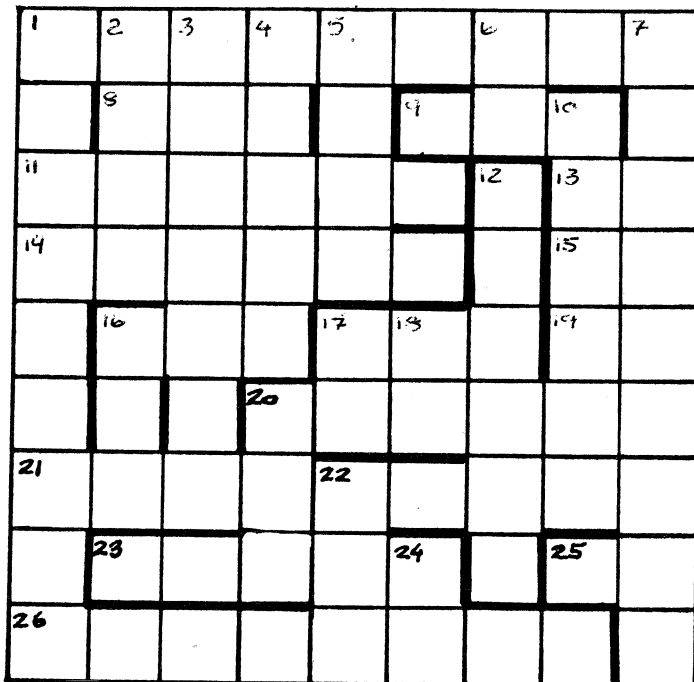
Eirik Lien, DMU-formann:

"Uff - enda et halvt år igjen av funksjonstiden"

Begge bidragsyterne belønnes med en pose Twist!



DATA-KRYSS



VANNRETT

- 1 Populært spill
- 8 Lite tegn (omv.)
- 9 Vesentlig enhet
- 11 Huskeplass
- 13 Er ikke arrangert for 1 vannrett
- 14 Trengs av 9 vannrett
- 15 Tall (omv.)
- 16 Måles magnetbånd i
- 17 Er U-1108 nå
- 19 Tall (langt i sør)
- 20 Moderne språk
- 21 Grenseland i dataspråk
- 23 Er dumpen
- 25 Også en retur (fork.)
- 26 Trenger de som skal ut på datafiske

Løsning sendes RUN-NYTT red., RUNIT innen 15. januar 1981.

Premiering!!

NAVN.....
ADR.....

LODDRETT

- 1 Mer enn en variabel
- 2 By
- 3 Irriterer terminalbrukeren
- 4 Bråstopp
- 5 Føles etter 4 loddrett
- 6 Har makt (fork.)
- 7 Oversette
- 10 Lager 9 vannrett
- 12 Godt å ha i reserve (omv.)
- 16 Sluttord
- 17 Tillate
- 18 Kald mat
- 20 Kjæledegge for teknokrat (se RUN-NYTT nr. 3 1979)
- 22 Sky (omv.)
- 24 Kan også datafolk