

J. 4

PSYKOLOGISK
PÆDAGOGISK
PÅ
DLH
FEBRUAR
1970

INDFØRING I BLOKOL - I
- om blokprogramsystemet,
fremstilling af databånd
og brug af GIER-datamater

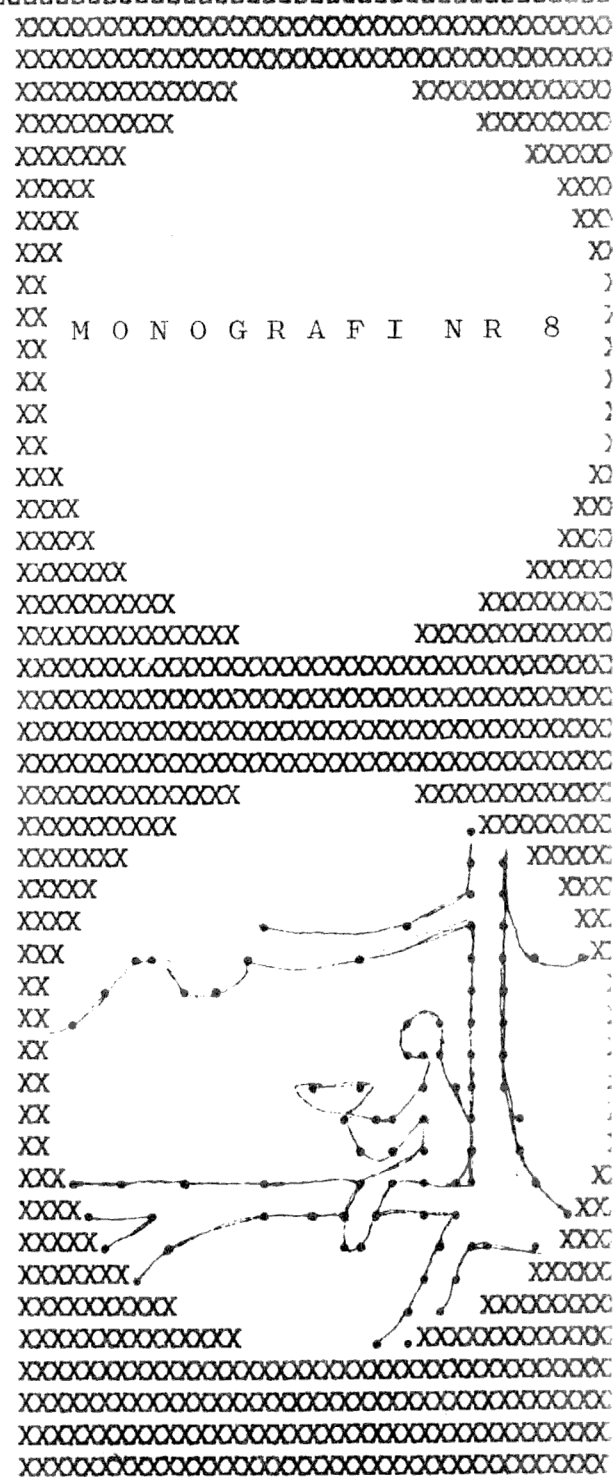
MONOGRAFI NR 8

OLE DREIER

PSYKOLOGISK - PÆDAGOGISK
VÆRKSTED

DATALOGISK INSTITUT
H. C. ØRSBØD INSTITUTET
UNIVERSITETSPARKEN 5
2300 KØBENHAVN Ø - DANMARK

INSTITUT FOR
INFORMATIONSTEKNOLOGI
UNIVERSITETSPARKEN 5
2300 KØBENHAVN Ø - DANMARK



Indholdsfortegnelse.

Indledning	s. 2
1. del. <u>Om blokprogramsystemet</u>	.
A. Om automatisk databehandling.	s. 2
B. Om baggrunden for blokprogramsystemet.	s. 3
BI. Om fælles træk ved data.	s. 4
BII. Om fælles træk ved operationer.	s. 6
C. Om blokprogramsystemet.	s. 7
CI. Om computer-programmer.	s. 7
CII. Om blokprogrammer.	s. 7
CIII. Om Blokøl.	s. 7
CIV. Om datablokke.	s. 9
CV. Definition af begreber i Blokøl.	s. 12
2. del. <u>Om fremstilling af databånd.</u>	
A. Om hvordan man blokstrukturerer et datasæt.	s. 14
B. Om databeskrivelsen	s. 15
BI. Om hvad en databeskrivelse er	s. 15
BII. Om hvordan man udfylder en databeskrivelse.	s. 15
BIII. Om databeskrivelse via skrivemaskine.	s. 20
C. Om hulning.	s. 20
CI. Om det lille tastatur.	s. 20
CII. Om når man huller forkert.	s. 22
CIII. Om strimlerne.	s. 23
CIV. Om dobbelthulning.	s. 23
3. del. <u>Lidt om brug af GIER-datamater.</u>	
A. Om data og programmer.	s. 24
B. Om datamatens kommunikationsmedier (ydre enheder).	s. 24
C. Om programsproget Algol 4.	s. 27
D. Om monitorsystemet HELP 3.	s. 27
E. Om strukturen i en kørsel.	s. 28
Litteratur	s. 29

Indledning.

Denne monografi er tænkt som første del af en revideret beskrivelse af blokprogramsystemet. Der foreligger på nuværende tidspunkt - februar 1970 - en del publikationer om disse programmer, men for alle undtagen en gælder det, at systemet er revideret så kraftigt siden deres fremkomst, at reviderede beskrivelser er nødvendige. Den undtagne publikation er: Jan Rattleff: - Et år med Blokøl - SKETs monografiserie 7. Duplikeret. Danmarks Lærerhøjskole. Kbhv.1969. Nævnte publikation er ikke en så bred beskrivelse, som den serie af beskrivelser, der påbegyndes med denne monografi, men kan snarere betragtes som et baggrunds- eller kildemateriale for store dele af det, der beskrives i den hermed påbegyndte serie af beskrivelser. (Jan Rattleff har assisteret ved udarbejdelsen af dele af manuskriptet).

1. del: Om Blokprogramsystemet.
- A. Om Automatisk databehandling

Indenfor forskningen har datamaterne tre væsentlige anvendelsesområder:

1. Styring: Datamaterne kan bruges til at styre og kontrollere gangen i videnskabelige forsøg.
2. Simulering: Konstruktion og afprøvning af videnskabelige modeller, således at aspekter ved computerens symbolske adfærd simulerer eller imiterer begivenheder i virkeligheden.
(Se f.eks. Abelson, R.P.: Simulation of Social Behavior - i Lindzey, G. og Aronson, E.: - The Handbook of Social Psychology - Addison-Wesley. Massachusetts 1968. Vol.2,p. 274 - 356.)
3. Behandling: Automatisk behandling af data udføres
 - a. for at teste opstillede hypoteser eller teorier
 - b. for nærmere at udforske sider ved virkeligheden, som er repræsenteret i de indsamlede data.
 - c. for at træffe beslutninger (f. eks. i form af diagnoser) om de objekter, personer eller hændelser, der er repræsenteret i de indsamlede data.

Det er udelukkende databehandling, der er emnet for skrifterne om blokprogramsystemet, fordi systemet på nuværende tidspunkt udelukkende anvendes til det formål.

En forskers motiv til at lade sine data behandle af en datamat er: hurtigere og sikrere end det kunne lade sig gøre med de traditionelle manuelle (evt. lidt automatiserede) metoder at få opfyldt et enkelt eller en kombination af de 3 ovennævnte formål (pkt. a, b og c).

For at det kan ske, må data imidlertid overføres til en datamattilgængelig form på et datamattilgængeligt medium: papirhulbånd, hulkort, magnetbånd. Desuden må de fejl, der opstår ved denne overførsel findes og fjernes fra det overførte datasæt.

B. Om Baggrunden for blokprogramsystemet.

I de sidste ca. 4 år har et team omkring amanuensis cand. psych. Jan Rattleff arbejdet med udformningen af et sæt af computerprogrammer. Målet for dette arbejde er:

1. at udarbejde en programsamling, der kan anvendes til **automatisk databehandling af alle de psykologiske datasæt**, der med fordel kan behandles automatisk.
2. at programsamlingen skulle være komplet i den forstand, at den skulle kunne udføre **alle de operationer med data**, som psykologer kunne ønske udført med deres data, og som med fordel kan udføres automatisk.

Systemet er efter min mening nået et godt stykke hen ad vejen mod disse mål, idet der nu foreligger en lang række ret generelle programmer udviklet i GIER Algol 4. Det konkrete programmeringssprog, som programmerne er udviklet i, sætter nogle begrænsninger på anvendelsesmulighederne af programmerne i deres nuværende form: De kan kun anvendes på GIER-datamater, der er udstyret med monitorsystemet Help 3 og programmeringssproget Algol 4. Et sådant arbejde med udformningen af et generaliseret sæt af computerprogrammer til databehandling af resultaterne af psykologiske undersøgelser kan kun være umagen værd og frugtbart, hvis der er mange **fælles træk mellem** de forskellige psykologiske **datasæt**, der ønskes behandlet, **og** hvis der er mange **fælles træk ved de operationer**, der ønskes udført med disse data. Er det nemlig tilfældet, vil et begrænset sæt af programmer kunne klare **alle** forefaldne opgaver, og man har da sparet en mængde af det meget tidkrævende og kostbare programmeringsarbejde.

Ved at analysere strukturen i en række konkrete psykologiske datasæt og ved at analysere de former for behandling af disse data, som psykologer ønsker udført, er vi kommet til den opfattelse, at der er mange fælles træk ved såvel data som de operationer, der ønskes udført dermed, så at det altså vil være frugtbart at søge et generaliseret programsæt udviklet.

Vi vil kort referere disse fælles træk, først ved data.

B I. Om fælles træk ved data.

Mange psykologiske eksperimenter, hvori der indgår målinger af den ene eller anden art, kan karakteriseres ved, at vi har nogle personer og nogle påvirkningskræfter i situationer og nogle reaktionsmåder. At det er tilfældet kan udnyttes til at skabe en fælles struktur i de psykologiske data.

Allerede selve de eksperimentelle designs er for det meste struktureret omkring de tre dimensioner, men endnu tydeligere bliver de, når psykologen registrerer sine data fra eksperimentet (på prøveark, i forsøgsprotokoller o.a.). De lovmæssigheder, psykologen synes at være interesseret i at finde, kræver åbenbart ofte data struktureret omkring de 3 dimensioner.

Endelig tilpasses data undertiden ved omformninger i endnu højere grad til de 3 dimensioner, når psykologen skal til at behandle sine data.

1. Hoveddimensionen synes at være personer. Det ses tydeligt i de data, psykologen har registreret. Det er her altid muligt at identificere, fra hvilken person et givent datum stammer. Strukturen i de registrerede data er oftest sådan, at man først har alle data fra person nr. 1, dernæst fra person nr. 2, osv. Oftest er personen ikke blot karakteriseret ved et nummer eller navn, men også ved en række yderligere informationer, der er uafhængige af selve eksperimentet: biografiske data såsom køn, alder, klasse, erhverv samt oplysninger om, hvilken kategori af forsøgspersoner denne person tilhører (såsom hvilken forsøgsgruppe). Alle disse informationer har en overordnet betydning i forhold til påvirkningskræfter og reaktionsmåder, hvilket tydeligt ses af de operationer med data, som psykologen ønsker udført.

2. Påvirkningskræfterne registreres ikke altid direkte. De er tit

givet i og med registreringen af en reaktion og denne reaktions placering i forhold til de øvrige registrerede reaktioner fra samme person. Det kan f. eks. være, fordi alle personer udsættes for en række påvirkningskræfter i samme rækkefølge, eller fordi man har en række påvirkningskræfter, hvis præsenteringsrækkefølge ikke kendes eller er ligegyldig, og man da vælger at repræsentere dem i en og samme rækkefølge for alle personer. Men undertiden må man ty til kort at identificere påvirkningskræften sammen med angivelsen af reaktionen derpå.

3. Reaktionsmåderne. Antallet af registrerede reaktionsmåder for hver påvirkningskraft kan variere fra nul og op efter. Der er det fælles træk ved data, hvori der indgår målinger af den ene eller anden art, at reaktionsmåderne kan inddeles i kategorier. Det medfører, at de kan repræsenteres med simple symboler som tal, bogstaver eller enkelte ord. Det gælder både når reaktionerne er kvalitative: ja eller nej eller forkert eller rigtigt på et spørgsmål eller i en opgave eller en af flere mulige reaktioner i en Rorschachtest, og når de er kvantitative: reaktioner der varierer i størrelse som placeringer på (nogle former for) skalaer. Det er dog ikke alle psykologiske undersøgelser (og heller ikke altid undersøgelser fra andre fagområder) der er bygget op omkring de tre dimensioner, eller der har de tre dimensioner i samme indbyrdes strukturelle forhold. Men i mange af disse tilfælde vil strukturen i data også naturligt være eller kunne omdannes til at være tredimensional. Og ydermere vil den ene dimension da også tit have en vis overordnet funktion i forhold til de to øvrige. Det kan f.eks. være en forsøgssituation, der er den overordnede dimension, eller en gruppe eller en tidsenhed i et forløb. Det betyder, at også disse datasæt ofte vil kunne databehandles af blokprogrammerne.

Disse fælles træk ved psykologiske data fra alle de undersøgelser, hvori der indgår målinger af den ene eller anden art, gør det muligt også at skabe en fælles struktur i data, når de bliver omskrevet til et datamat-tilgængeligt medium: transkriberende, hullede, kodede, renskrevne. Det gøres ved at inddele data i blokke. Før vi går nærmere ind herpå, skal vi dog beskæftige os lidt med de fælles træk ved de operationer, der ønskes udført på data.

B II. Om fælles træk ved operationer.

På side 2 anføres, hvilke formål en databehandling af videnskabelige datasæt kan have for forskeren. Det antydes imidlertid også, at databehandlingen må omfatte andre faser end selve testningen af opstillede hypotecer etc. De Operationer der indgår i en databehandling af et datasæt kan således inddeles i følgende faser:

1. Fremstilling af databånd (ofte kaldet hulning). De registrerede data skal først omformes og overføres til et datamattilgængeligt medium som f. eks. hulbånd.
2. Trimning. Herved forstås operationer hvorved fejl findes i og fjernes fra de indhullede data.
3. Reorganisering. Disse operationer har til formål at reorganisere data på databåndet. En sådan reorganisering er som regel ikke noget formål i sig selv, men er hjælpeoperationer for
 - a. at tilpasse data til bestemte operationer i andre faser, især de to følgende
 - b. at organisere data således at man ved at tage en udskrift af dem lettest kan danne sig et overblik over eller indblik i dem evt. manuelt udføre beregninger udfra udskriftterne.
4. Tabellering. Herved forstås operationer hvorved aspekter ved data opstilles i tabeller med 1, 2 eller 3 dimensioner og hvorved data i de enkelte celler eller dimensioner sammentæles. Disse operationer er principielt de samme som man betegner med ordet "olregnskab", men da de fleste forskere ville reagere på at få deres behandling af videnskabelige data betegnet som "olregnskab" ved desperat at søge udtænkt andre operationer, som de istedet ville udføre, og som de ikke ville vide nok om til at udføre forsvarligt, har vi undgået at bruge denne betegnelse som overskrift.

Operationerne i denne fase er ofte de sidste i en databehandling af et givet materiale, da de eventuelle statistiske tests, der yderligere måtte skulle til for at teste hypotecer etc., ofte er forskelligartede, at det mød fordel kan overlades til den enkelte forsker selv at udføre dem på grundlag af de tabellerede resultater.
5. Statistisk bearbejning. Herved forstås operationer med data på grundlag af statistiske modeller. Som ovenfor nævnt er det

ikke alle datasæt, der ønskes udført sådanne operationer med, og ønskes de udført, er de oftest så forskelligartede og tager yderligere så kort tid, hvis data er behandlet automatisk i de foregående faser, at de med fordel kan udføres manuelt. Men undertiden er der behov for statistisk bearbejdning, der med fordel kan udføres automatisk. Det gælder f.eks. beregning af korrelationskoefficienter og item analyse.

C. Om blokprogramsystemet.

C I. Om computer-programmer.

De operationer, som man ønsker, at en datamat skal udføre for en, skal meddeles datamaten. Det gøres i programmer. Et program er en hierarkisk organiseret sekvens af instruktioner om elementære operationer, som man ønsker, at datamaten skal udføre (en lidt mere udførlig behandling af datamatens funktionsmåde gives på side 24). Også programmerne skal omskrives til en datamattilgængelig form og meddeles på et datamattilgængeligt medium, i vort tilfælde hulbånd. De består groft sagt af instruktioner om at indlæse data, om at udføre visse operationer med disse data og om at outputte resultaterne af disse operationer.

C II. Om Blokprogrammer.

Som omtalt i de forrige afsnit er der fælles træk ved psykologiske datasæt og ved de operationer, der ønskes udført med disse data. Det gør det muligt at standardisere de instruktioner, datamaten skal have, om at udføre elementære operationer med data. Det gør det muligt at udføre et standardiseret sæt af programmer. Disse programmer er de såkaldte blokprogrammer.

Blokprogrammer er programmer, der kan udføre de operationer, der indgår i dem, på alle blokstrukturerede materialer, det vil ifølge det foregående sige: alle psykologiske datasæt hvori der indgår målinger af den ene eller anden art. Der er nu udarbejdet et sæt af sådanne programmer, så både trimning, reorganisering, tabellering og statistisk bearbejdning nu kan udføres med blokprogrammer.

C III. Om Blokøl.

Det viste sig, at blokprogrammerne havde mange fælles træk i opbygning og indhold, så at man kunne sige, at de var skrevet i deres eget karakteristiske sprog. Dette sprog kaldte vi Blokøl. De fælles operationer trak vi nu ud af de enkelte programmer og dan-

nede et særligt overordnet program af dem og kaldte dette for blokblokken.

Dette program rummer følgende fælles funktioner (jvf. - Et år med Blok1 -):

1. at man kan manipulere direkte og på en standardiseret måde med blokbegreberne (se om disse side 12).
2. at man kan automatisere og standardisere administration af datamatens ydre enheder og af fordelingen af lagring mellem datamatens baggrundslager og arbejdslager (se side 27).
3. at man kan standardisere kommunikationen mellem datamaten og operatoren af datamaten.
4. at man kan foretage en standardiseret automatisk afmærkning af output (jvf. side 16).

Udover de store fordele som de ovennævnte fire punkter i sig selv indebærer for udvikling og ændring af programmer og for korse- len med disse programmer, har udarbejdelsen en af blokolsproget og blokblokprogrammet følgende to fordele:

5. Udarbejdelsen af nye programmer og revisioner af eksisterende programmer er blevet meget lettere. Det er af ret stor betydning, fordi netop disse faser af en databehandling ville tage det meste af den tid, der ville gå med databehandlingen af et givet materiale, og samtidig fordi netop denne fase er et meget vanskeligt og krævende arbejde. Alle nye blokprogrammer kan nu blot bygges op omkring det skelet, som blokblokken udgør, og alle de fælles operationer i blokprogrammerne bliver nu inkorporeret i de nye programmer blot ved i disse at henvise til ("kalde") blokblokken. Selve blokprogrammerne kommer derved til at fylde meget mindre, og det sammen med den yderligere strukturelle opdeling af operationerne, som Blok1 indebærer, medfører at programmeringsarbejdet bliver meget mere overskueligt og dermed også betydeligt lettere, hurtigere og sikrere. Peter Naur udtrykker det således: "... systematik og simpelt, gode principper, må være afgørende, fordi en ting er ikke effektiv med mindre den er pålidelig, og en ting er ikke pålidelig med mindre man forstår den, og man forstår den ikke med mindre den er simpel." (Peter Naur 1966). Selvom størstedelen af programmeringsarbejdet i Blok1-programserien:

formodentlig er udført, så har netop skabelsen af Blokøl betydning overordentlig meget for, at Programmeringsarbejdet og systemet er kommet så langt, som det er, dels ved at sanere de programmer, der fandtes før Blokøl blev skabt, og dels altså også ved at lette udbygningen af programsættet.

6. Kørselen af programmerne på datamaten er blevet meget lettere, først og fremmest fordi kommunikationen mellem datamat og operator er blevet standardiseret. Man behøver altså ikke kende så meget til et bestemt program for at kunne betjene det, hvilket bl. a. også betyder, at andre end programmøren selv kan finde ud af at betjene det.

Ved alle kørsler med Blokøl-Programmer indlæses først det overordnede program, blokblokken, og dernæst det konkrete underordnede blokprogram, som man ønsker kørt. Blokblokken er ikke i sig selv noget helt og fuldstændigt program i den forstand, at det ikke selv kan udføre operationer med data. Hvilke operationer, der ønskes udført, skal altid defineres af det enkelte blokprogram. Først når dette også er indlæst, kan maskinen udføre de ønskede operationer med data. En mere detaljeret beskrivelse af, hvordan man kører med Blokøl-programmer, gives f. eks. i Ole Dreier: - Indføring i Blokøl 2: Bloktrimning -.)

C IV. Om Datablokke.

De fælles træk ved data fra alle de psykologiske eksperimenter, hvori der indgår målinger af den ene eller anden art, gør det som nævnt i afsnittet om fælles træk ved data (side 4) muligt også at skabe en fælles struktur i datarepræsentationen på hulbåndet: Data inddeles i blokke. Vi skal nu gå nærmere ind på, hvad en blok er.

En Blok.

En blok består oftest af alle data fra en bestemt forsøgsperson. Et databånd består altså af en sekvens af blokke.

Blokkene har imidlertid også deres egen indre struktur:

Blokhovedet.

Første afsnit af blokken kaldes blokhovedet. Det består af alle informationer om selve personen. Det kan være oplysninger som: forsøgspersonnummer, køn, alder, klasse, erhverv, forsøgspersongruppe etc. Det fælles ved disse oplysninger er,

at de tit har en overordnet funktion i forhold til de øvrige data om denne person (heraf navnet blokhoved). Det ses af, at behandlingen af informationerne i den resterende del af blokken styres af informationerne i blokhovedet: en reaktion i et forsøg behandles f.eks. tit forskelligt afhængigt af, fra hvilken person den stammer, herunder personens alder, køn, erhverv, klasse, forsøgspersongruppe etc. Dette gælder reorganiseringssfasen, tabelleringsfasen og den statistiske bearbejdningsfase.

Blokkroppen.

Den resterende del af blokken udgør blokkroppen. Her står angivet en persons reaktioner på påvirkningskræfterne samt eventuelt en identifikation af påvirkningskræfterne. Som omtalt i afsnittet om fælles træk ved data (side 4) er det ikke altid nødvendigt eksplicit at repræsentere påvirkningskræfterne, da de tit er givet implicit.

Kroppen inddeles i helheder og disse igen i dele.

Dele: En del består af en enkelt information, f.eks. personens score i en bestemt opgave.

Helheder: Men undertiden kan man med fordel inddele blokkroppen i større og mere overordnede enheder end delene. Det er f. eks. tilfældet, når en persons reaktion på en enkelt påvirkningskraft er karakteriseret ved mere end en oplysning (som f. eks. både score og tid for en given opgave), eller når det udførte eksperiment naturligt kan inddeles i mere overordnede enheder end de enkelte reaktioner og påvirkningskræfter (situationer) som når det f. eks. består af flere delforsøg med hver deres sæt af reaktioner. Er det tilfældet, siger man, at kroppen består af flere helheder. Er det derimod ikke tilfældet, består den altså kun af 1 helhed, der så eventuelt kan have flere dele.

Ved at anvende en sådan tottrins inddeling af kroppen opnår man en stor smidighed. Mange datasæt falder naturægt ind i den. Andre består kun af et trin. Man kan da udnævne dette trin til udelukkende at være helheder f. eks., dvs. at hver helhed kun har en del, eller man kan udnævne det til kun at

være dele, dvs. at kroppen består kun af en helhed med en mængde dele i.

Endnu to ting bør nævnes om strukturen i kroppen. For det første kan man undertiden have en tom krop, dvs. en blok der kun består af et blokhoved, fordi ens data er så enkle, at man har set en fordel i at anbringe dem alle i blokhovedet. Antallet af helheder i kroppen bliver da pr. definition nul. Og for det andet skal antallet af dele i en helhed være det samme i alle helheder i et givet datasæt.

Der skal nu gives et eksempel på en blokstruktur for et psykologisk datasæt.

a1,2,080453,211169,10

13,1

27,1

35,1

56,2

72,2

a2,1,090353,211169,10

17,1

21,1

osv.

De data, der er blevet indhullet i disse blokke, kunne f.eks. være følgende:

Person nr 1. pige fødselsdag, prøvedag, klasse

opg.1, tid: 13 sek., besvaret rigtigt

- 2, - 27 - , - -

- 3, - 35 - , - -

- 4, - 56 - , besvaret forkert

- 5, - 72 - , - -

person nr. 2, dreng, fødselsdag, prøvedag, klasse

opg. 1, tid: 17 sek., besvaret rigtigt

- 2, - 21 - , - -

Det ses, at alle data repræsenteres ved simple symboler, og at symbolerne er entydige: forskellige symboler for forskellige oplysninger (jvf. side). Det ses endvidere, at hver blok begynder med et særligt symbol, blokstartsymbolet, i dette tilfælde bogstavet a.

C V. Definition af begreber i blokol.

Der skal nu gives en definition af de begreber i blokol, som man må have kendskab til for at kunne benytte systemet. Der begyndes med de mest omfattende begreber.

Datasæt: Et datasæt består af et eller flere delsets.

Delsæt: Et delsets kan begynde med en delsetsoverskrift og slutter med et end. Det består af en eller flere blokke.

Delsætsoverskrift: En delsetsoverskrift er alt indtil første start i et delsets (begrebet er indført, for at man der kan have informationer om, hvad det er for et materiale, der er i sættet, og eventuelt få disse informationer overført til outputtet).

End: End defineres i programmerne som stopkode (flexowriterkode 11) og betyder, at et delsets afsluttes.

Blok: En blok består af et starttegn (start), et hoved og evt. en krop, samt tre indbyrdes forskellige skilletegn: lineskift, helstop (punktum) og delstop (komma).

Start: Starttegnet er et bogstav, der indleder en blok (for at give redundans).

Hoved: Blokhovedet indeholder et antal (inf) heltal (fra 1 til 39 tal) adskilt med skilletegnet delstop. Det sidste tal i hovedet afsluttes dog med skilletegnet lineskift eller helstop.

Krop: En krop indeholder et antal helheder (fra 0 og opefter), der igen består af samme antal dele (fra 1 og opefter). Hver helhed afsluttes med skilletegnet helstop eller lineskift (den sidste helhed i kroppen dog altid med lineskift). Hver del adskilles indbyrdes med skilletegnet delstop. (Det ses af disse definitioner, at en blokkrop kan indeles udelukkende i dele: antallet af helheder går helt ned til 0 og antallet af dele fra 1 og opefter.

Stormax: Stormax er det maksimale antal helheder, en blok i et givet datasæt kan have.

Stor: Stor er det konkrete antal helheder, som en given blok i et givet datasæt har.

Fast: **Formatet** i et datasæts blokstruktur siges at være fast, hvis alle blokkenes stor = stormax, dvs. hvis alle blokke i datasættet har samme antal helheder.

Variabelt: Formatet i et datasæts blokstruktur siges at være variabelt, hvis blokkenes antal af helheder varierer. Er formatet i datasættet variabelt, skal det sidste tal i blokhovedet angive stør, altså antallet af helheder i denne konkrete blok.

2. del: Om fremstilling af databånd.

A. Om hvordan man blokstrukturerer et datasæt.

For at kunne få behandlet sine data på en datamat, må de overføres til hulbånd (i andre tilfælde bruges hulkort eller magnetbånd). På et hulbånd repræsenteres hvert enkelt tegn (de tegn der findes på en flexowriter) med en unik hulkombination.

Hver hulkombination har sin egen talværdi. Koden herfor kaldes flexowriterkoden og kan slås op i - A Manual of GIER Algol 4 - side 7. Også bogstaver repræsenteres her med en talværdi, bogstavet a har f. eks. flexowriterkoden 49.

Et hulbånd består da af en række af sådanne hulkombinationer efter hinanden.

Til en sådan struktur skal data overføres, og ydermere overføres så tegnene på hulbåndet står i en blokstruktur.

Hvad der forstås ved en blokstruktur, og altså hvordan data skal repræsenteres på hulbåndet, er omtalt i afsnittet om datablokke, (side 9).

Af programmeringsmæssige hensyn bør nummereringen af blokkene være fortløbene.

Der er endnu 3 regler, som henholdsvis bør og skal overholdes, når man blokstrukturerer sit materiale.

For det første bør hver enkelt information repræsenteres med så simpel en symbolstreng som muligt.

Og for det andet skal repræsentationen af en given information være entydig: den må kun kunne betyde en ting. Entydigheden gælder for hver oplysning for sig. Man kan altså godt repræsentere oplysninger forskellige steder i blokken (f. eks. en persons nummer og en oplysning i blokkroppen) med samme symboler (f.eks. talrækken).

Og endelig for det tredje: Anvendes der tal i repræsentationen, kan man ikke repræsentere dem som decimaltal, men udelukkende som heltal. Det skyldes, at komma og punktum som tidligere nævnt anvendes som skilletegn mellem de enkelte informationer i blokken. Man ville altså bryde reglen om entydighed ved også at bruge dem som decimaltegn. Problemet klares dog forholdsvis let ved under hele databehandlingen at repræsentere eventuelle decimaltal som heltal og så bagefter dividere resultaterne med en

passende størrelse, eller ved at indføje denne division i programmerne. Man må blot huske, at repræsenterer man f. eks. tiden som tiendedele sekunder i en blok, så må de også repræsenteres som sådanne i alle andre blokke. Man kan også løse problemet ved i programmerne at omdefinere skilletegnenes værdi, hvis man på denne måde kan frigøre komma eller punktum og anvende et andet tegn (der da udelukkende kan anvendes i denne betydning) som skilletegn.

Endelig må det bemærkes, at hvis der i de registrerede data mangler informationer, som skal med på hulbåndet (f. eks. forsøgspersonnummer), tilrådes det at indføre disse på registreringsarkene før indhulningen, så at hulningen kan forløbe uden unødige afbrydelser og så automatisk som muligt.

B. Om databeskrivelsen.

B I. Hvad en databeskrivelse er.

En række af de egenskaber, der er fælles for alle blokstrukturerede materialer, varierer i værdi fra materiale til materiale. Det gælder f. eks., hvor mange heltal der er i blok hovedet. For at kunne behandle de forskellige materialer må blokprogrammerne have defineret værdierne af disse variable i det konkrete materiale, der ønskes behandlet. Det kunne gøres fra kørsel til kørsel af operatoren på datamatens skrivemaskine, men gøres lettere (og mere fejlfrit) ved en gang for alle før man begynder at køre dette materiale at udarbejde en databeskrivelse af materialet. I denne defineres altså de konkrete værdier, som en række variable egenskaber ved blokkene antager i dette materiale. Den hules ind på hulbånd, så at operatoren blot behøver at lægge den i datamatens læser, hver gang han kører med materialet, istedet for at skulle kunne huske alle variablenes værdier i pågældende materiale. På den måde kan også operatører, der ikke kender til materialet, køre med det.

B II. Om hvordan man udfylder en databeskrivelse.

For at lette udarbejdelsen af databeskrivelsen er der udarbejdet et bånd med en ikke udfyldt databeskrivelse på. Det sætter man i en flexowriter, slår puncheren til og kopierer frem. Båndet er

forsynet med en stopkode, så det stopper kopieringen, hver gang man skal udfylde. Det sættes igang igen ved at trykke på "start read" på flexowriteren. Til **sidst** må man huske at forsyne den færdige databeskrivelse med en stopkode. De stopkoder, der findes på den ikke udfyldte databeskrivelse kopieres ikke med. Nedenfor er den ikke udfyldte databeskrivelse afbildet. Stopkoderne på båndet er her erstattet med å'er, det vil altså sige, at alle de steder, hvor der står et å, skal der indføres en beskrivelse.

Databeskrivelse til BLOKOL for flg. materiale: å

Udarbejdet af: å/å;

kor å

start å

inf å

fast å

Stormax å

dele å

tegn å

hoved

under å

over å

speciel å

dele

under å

over å

speciel å

type å

Vi skal nu gennemgå, hvorledes man udfylder databeskrivelsen:

1. Først skriver man en betegnelse der kan bruges til at identificere materialet.
2. Dernæst skrives kort af hvem og hvornår beskrivelsen er udarbejdet første gang.
3. En betegnelse for af hvem og hvornår beskrivelsen sidst er revideret.
4. En betegnelse for, hvilket materiale det er. Den vil komme med på alle output på skrivemaskinen, så man bagefter kan se hvor ens kørsel begyndte, og hvilket materiale man kørte.

Desuden overføres de første 6 tegn af den til en variabel (NAVN) i blokblokken, og kan derfor udskrives i det egentlige output fra kørslen.

5. En betegnelse for, hvilket blokstartsymbol der er anvendt. Symbolets værdi skrives i flexowriterkode (se - A Manual of GIER Algol 4 - side 7).
6. Antallet af heltal i blokhovedet.
7. Om antallet af helheder i blokkroppen er konstant for alle blokke i datasættet. Hvis skrives 1, hvis ikke 0.
8. Det maksimale antal af helheder i en blokkrop i dette datasæt.
9. Hvor mange dele der er i hver helhed i blokkroppen.
10. Hvor mange forskellige tegn, der er brugt i blokkroppen (eksklusiv skilletegnene).
11. Den laveste værdi et tal i blokhovedet må antage. Det skrives for hvert af tallene i rigtig rækkefølge, og efter hvert tal skrives et komma.
12. Den højeste værdi et tal i blokhovedet må antage. Det skrives for hvert af tallene i rigtig rækkefølge, og efter hvert tal skrives et komma.
13. En eventuel speciel værdi for hovedets tal, altså **en værdi**, der ligger udenfor området under - over. Findes ingen sådanne skrives blot en værdi indenfor. Der skrives et tal for hvert heltal i blokhovedet i rigtig rækkefølge, og efter hvert tal skrives et komma.
14. Afhængigt af hvilken type en del i kroppen har (anføres under pkt.17), altså om en given del er af typen tal, **streng eller tegn**, , angives her for hver del i kroppen i rigtig rækkefølge og efterfulgt af komma enten a) det laveste tilladte tal i den del eller b) det laveste tilladte antal tegn i denne del eller c) den laveste tilladte værdi i den del ifølge flexowriterkoden.
15. Afhængigt af hvilken type en del i kroppen har (anføres under pkt. 17), altså om en given del er af typen tal, streng eller tegn, , angives her for hver del i kroppen i rigtig rækkefølge og efterfulgt af komma enten a) det højeste tilladte tal i den del eller b) det højeste tilladte antal tegn i den del eller c) den højeste tilladte værdi i den del ifølge flexowriterkoden.

16. Afhængigt af hvilken type en del i kroppen har (anføres under pkt. 17), altså om en given del er af typen tal, streng eller tegn, . angives her for hver del i kroppen i rigtig rækkefølge og efterfulgt af komma enten a) en eventuel speciel talværdi eller b) et eventuelt specielt antal tegn i den del eller c) en eventuel speciel værdi i den del ifølge flexowriterkoden. Findes der ingen værdier udenfor de områder, der er angivet under pkt. 14 og 15, skrives blot en værdi i dette interval.
17. Her anføres for hver del i kroppen i rigtig rækkefølge og efterfulgt af et komma, om symbolerne i delene er a) tegn: både tal og bostaver, der skrives -1, eller b) heltal, der skrives 0, eller c) **streng**, der skrives +1.
18. Endelig anføres på linien nedenfor (det kan man ikke se på udskriften af den ikke udfyldte databeskrivelse) hvilke tegn der er anvendt (er lovlige) i blokkene. Først skrives de anvendte tal, dernæst de anvendte bogstaver. Der skrives intet skilletegn mellem dem.

Som eksempel på en udfyldt databeskrivelse for et materiale gives følgende:

Databeskrivelse

kulør præference efter konvertering/ijl10270;

kør kulør præference

start 49,

inf 16,

fast 1,

størmax 0,

dele 0,

tegn 0

hoved

under 1,0,0,0,1,1,1,1,1,1,0,0,0,0,0,0

over 812,6,9,2,7,7,7,7,7,7,197,197,197,197,197,197

speciel 1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0

Databeskrivelse til BLOKOL for flg. materiale:

kulør præference

Udarbejdet af: 1j/10.2.70.;

kør kulør præference

start 49,

inf 4,

fast 1,

størmax1,

dele 7,

tegn 10

hoved

under 1,0,0,0

over 812,6,9,2

speciel1,0,0,0

dele

under 123456,0,0,0,0,0,0

over 654321,197,197,197,197,197,197

speciel0,0,0,0,0,0,0

type 0,0,0,0,0,0,0

1234567890

B III. Om databeskrivelse via skrivemaskinen.

Har man ikke udarbejdet en databeskrivelse på bånd, kan programmerne køre ved, at operatoren taster nogle af de mest uundværlige informationer på datamaterns skrivemaskine. Blokblokken, der styrer indlæsningen af databeskrivelser fra hulbånd, administrerer også dette.

Det sker ved under kørslen at svare nej på blokblokkens spørgsmål om, hvorvidt databeskrivelsen er i læseren, og dernæst besvare dens spørgsmål om værdierne af variablene start, inf, fast og stormax. Disse tal tasteres på datamaterns skrivemaskine og efterfølges af et komma.

I trimningsfasen bruges informationerne i databeskrivelsen til at styre fejlsøgningen og definere, hvad der er rigtigt. Bruger man altså kun de 4 informationer, der kan tasteres på skrivemaskinen, vil fejlsøgningen blive mindre intensiv. I de andre faser af databehandlingen har de øvrige informationer i databeskrivelsen ikke stor betydning, men erfaringsmæssigt vil operatoren ofte enten ikke kunne huske de fire variables værdier eller taste forkert på skrivemaskinen.

C Om hulning.

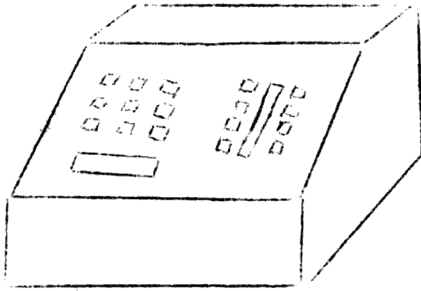
For at kunne benytte programmerne og datamatern må de registrerede data omskrives til en struktur på hulbånd.

Denne indhulning foregår traditionelt på en flexowriter, men der er på Danmarks Pædagogiske Institut udviklet et tastatur, som indebærer visse fordele.

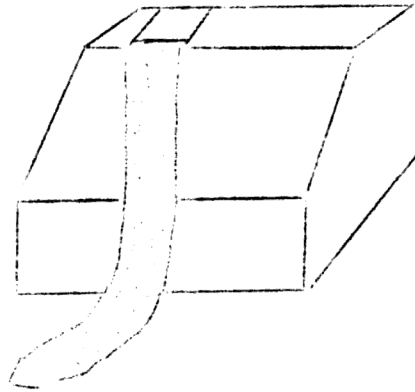
C I Om det lille tastatur.

Tastaturet er afbildet på næste side. Det har de fordele, at det er mere overskueligt end en flexowriter (der har et tastatur af størrelse omtrent som en skrivemaskine) og samtidig for det meste er tilstrækkelig stort, da man oftest repræsenterer sine data med et ret lille antal forskellige symboler.

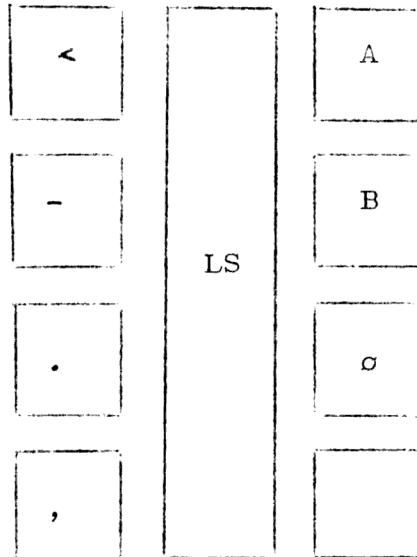
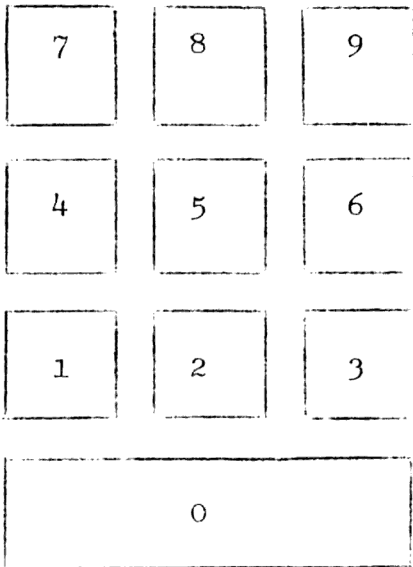
Anvendelsen af et mindre tastatur medfører færre fejlslag og hurtigere hulning. Endvidere kan man når man huller på et sådant tastatur, have selve perforatoren, der støjer ret ubehageligt (også på en flexowriter) stående i et tilstødende rum, hvilket er en stor arbejdspsykologisk fordel. Og endelig er der med det



tastatur



perforator



lille tastatur bedre plads til, at man kan have sine papirer liggende foran sig (prøvearkbunker o.l.), hvilket også er en arbejdspsykologisk fordel.

Det tilrådes at udarbejde et kort over tastaturet og sætte det op foran sig, når man huller. På kortet indfører man på hver knaps plads, hvilke tegn i de registrerede data, der skal repræsenteres på hulbåndet ved tryk på denne knap på tastaturet.

C II Om når man huller forkert.

For at gøre hulningen behageligere og først og fremmest for at lette rettelser af hullefejl, der opdages med det samme, er der i trimningsprogrammerne (det første program i denne serie, kaldet sletningsprogrammet, se - Indføring i Blok 2: Bloktrimning) indarbejdet nogle slettefunktioner. Slettefunktionerne betyder, at man kan benytte nogle knapper på tastaturet (eller flexowriteren) som rettetegn. Disse knapper må så naturligvis ikke samtidig have andre betydninger. Vi skelner mellem tre forskellige rettetegn:

1. Slettegn. Trykkes der på denne knap slettes det sidste hullede tegn fra strimmelen (af sletningsprogrammet). En række af slettegn betyder, at et tilsvarende antal hullede tegn slettes. Almindeligt anvendt som slettegn er tegnet "o".

2. Sletlinie. Trykkes der på denne knap, slettes alt til og med det sidst hullede linieskift. En række af sletlinietegn betyder, at et tilsvarende antal linier vil blive slettes fra hulbåndet. Almindeligt anvendt som sletlinie er tegnet "⌋".

3. Sletblok. Trykkes der på denne knap, slettes alt til og med sidste blokstartsymbol på hulbåndet. En række sletbloktegn vil føre til, at et tilsvarende antal blokke slettes. Almindeligt anvendt som sletblok er tegnet "◀".

Det må bemærkes, at slettegn også kan bruges til at slette et hullet sletlinie eller et sletblok, samt at sletlinie kan slette et sletblok.

Disse tegn kan ikke alene med fordel anvendes, når man ved, at man har hullet forkert, men også når man ikke er sikker på, om man hullede rigtigt, eller på hvor langt man kom med hulningen: man går tilbage til det sidste, som man ihvertfald er sikker på, at man har hullet.

C III Om strimlerne.

Enhver strimmel skal som sit første tegn have et strimmelstart-symbol. Som dette skal bruges flexowriterkode 11, altså stopkoden (forklaringen herpå gives under omtalen af sletningsprogrammets funktionsmåde i - Indføring i Blok 2: Om bloktrimning -).

Båndene rulles op på en papkerne med det først hullede inderst. Yderst på det oprullede databånd, som på alle databånd, skrives kort a) hvilket materiale det er (samme betegnelse som i kør i databeskrivelsen) b) hvem der har hullet det og hvornår c) hvilket personnummerinterval det indeholder.

De enkelte databånd kan sammenklistres med en speciel gul tape og en lille holder og klipper.

C IV Om dobbelthulning.

Alle datasæt skal hules ind to gange.

Den eneste forskel på de to versioner skal være værdien af start. Kan man ikke under hulningen give dem forskellige værdier, fordi man bruger alle tasterne på nær 1 til andre funktioner, kan man ændre værdien af start på det ene sæt efter hulningen, nemlig under kørslen med det første program de hullede data skal igennem: Sletningsprogrammet i trimningsfasen (se herom - Indføring i Blok 2: Om bloktrimning -).

Der skal hules to versioner, fordi man som led i trimningen tester indholdsmæssige fejl i det hullede ved at sammenligne med en anden parallelt hullet version af samme datasæt.

De to versioner bør hules af forskellige mennesker, fordi det er sandsynligere, at de samme fejl vil blive begået i begge hullede versioner, hvis de hules af samme person, og den indholdsmæssige kontrol, der gør brug af dobbelthulningen, gør netop dette ved at finde fejl, der ikke er begået i begge versioner.

Er der begået samme fejl i begge versioner, vil de ikke blive opdaget af dette program, men delvis af andre programmer.

3. del: Lidt om brug af GIER-datamaterne.

A. Om data og programmer.

Den der står og vil bruge en datamat til automatisk databehandling, ser formålet med dette i, at han har nogle data, som han gerne vil have datamaten til at udføre nogle operationer på. Disse formål vil i vort tilfælde være et af de på side 2 nævnte tre formål, dvs. for at teste hypoteser etc. Af afsnittet om fælles træk ved operationer (side 6) fremgår det imidlertid tydeligt, at man ikke blot umiddelbart kan lade datamaten udføre disse operationer, men først må hulle sine data ind, udføre fejlsøgning og rettelse etc.

For at datamaten kan vide, hvilke operationer, man ønsker udført med sine data, må det meddeles den. Det gøres ved at inddеле operationerne i enheder af passende størrelse og skrive dem i et program.

Ydermere gælder det for blokprogramsystemet, at programmerne er todelte. Det skyldes som tidligere omtalt, at de fælles operationer er trukket ud af de enkelte blokprogrammer og samlet i et enkelt overordnet program, blokblokken. Blokblokken indeholder imidlertid kun de fælles operationer og kan ikke i sig selv udføre nogen egentlig databehandling. Dertil kræves yderligere et blokprogram. Datamaten skal altså både forsynes med blokblokken og et blokprogram, hver gang man ønsker at køre et blokstruktureret materiale på den.

Og endelig gælder det for programmer i blokprogramsystemet, at de kræver en specificering af nogle variable ved det konkrete materiale, der ønskes kørt. Det sker i en databeskrivelse (se side 15).

B. Om datamatens kommunikationsmedier. (ydre enheder)

Datamaten skal have meddelelse om alt det foregående: hvilke data man har, hvilke operationer man vil have udført på dem (blokblokken og et blokprogram), og hvilke variable der skal specificeres (databeskrivelsen).

Disse meddelelser kaldes input: meddelelser fra operator til maskine.

Men maskinen skal også kunne give operatoren meddelelser, bl.a. resultatet af kørslen, f. eks. de behandlede data. Meddelelser fra maskinen til operatoren kaldes output.

Maskinen må altså have nogle ind- og outputmedier. Der er følgende:

1. Læser.
2. Skrivemaskine.
3. Perforator.
4. Linieskriver.
5. Plotter.

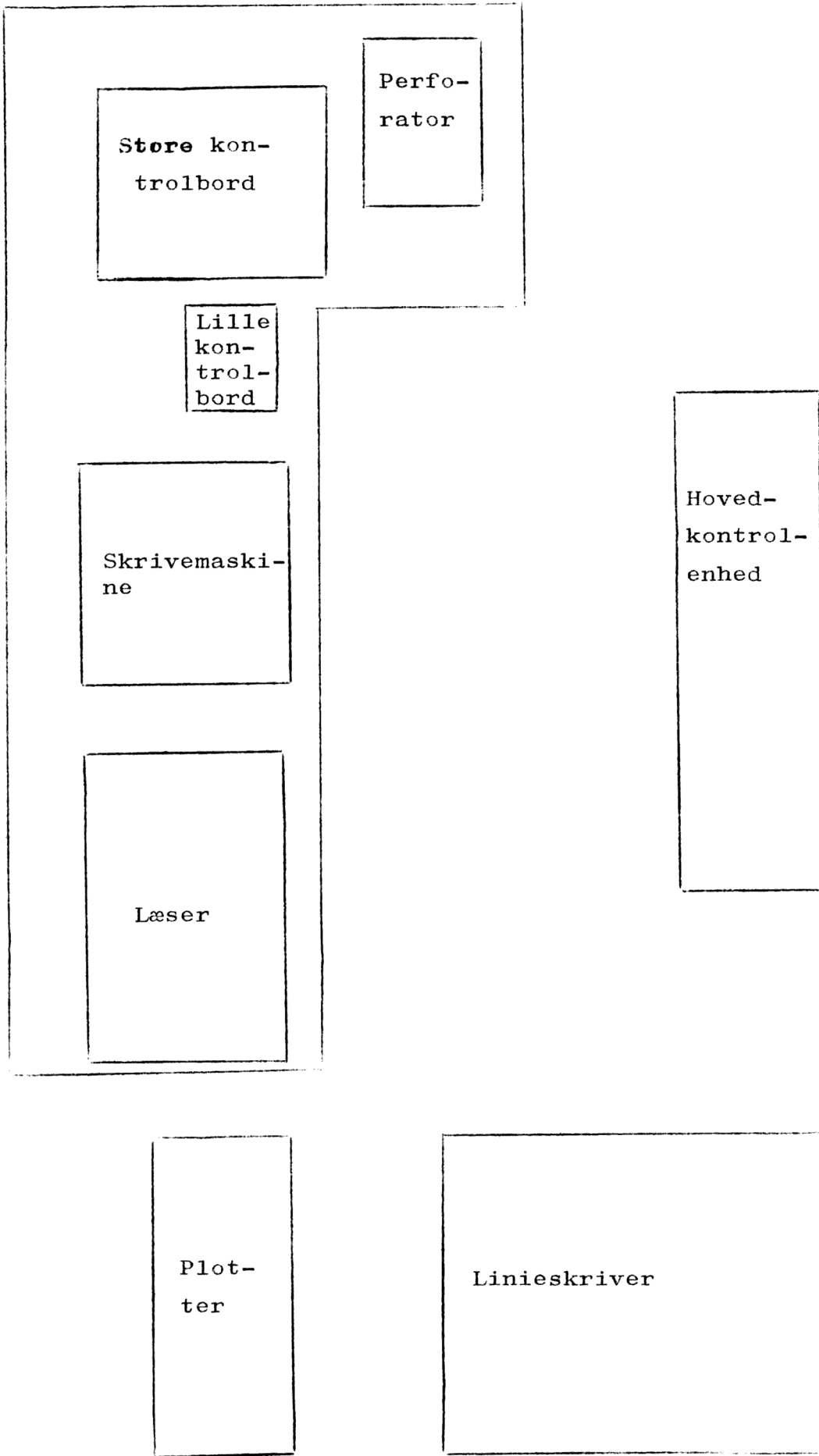
På næste side er der en skitse over datamaten med dens ind- og outputmedier.

Disse mediers funktion skal nu ganske kort beskrives.

1. Læseren. Den er datamatens vigtigste inputmedium. Igennem den inputtes alle hulbånd (programmer, data på bånd). Ved hjælp af et optisk system aflæses hulkombinationerne på båndene.
2. Skrivemaskinen. Over dette medium foregår størstedelen af den løbende kommunikation under kørslen mellem datamaten og operatoren.
3. Perforatoren. Den er nok datamatens vigtigste outputmedium. Her outputtes for det meste resultaterne af en kørsel.
4. Linieskriveren. (eller printeren). Her kan datamaten hurtigt outputte udskrifter i form af meddelelser til operatoren, udskrifter af resultaterne af en kørsel såsom tabeller.
5. Plotteren. Plotteren er et grafisk outputmedium.

Hvilke in- og outputmedier der bruges i en bestemt kørsel styres af programmet. Man kan imidlertid også på maskiner af type som GIER 2 på H.C. Ørsted ~~instituttet~~ styre fordelingen af ind- og output. Det sker på det lille kontrolbord til højre for skrivemaskinen. (se herom i Blok 1 II om bloktrimning).

Det må bemærkes, at samme output kan komme på flere medier samtidig, f. eks. linieskriver og perforator eller perforator og skrivemaskine.



C. Om programsproget Algol 4.

Kommunikationen med datamaten skal foregå i et bestemt sprog, som datamaten kan "forstå". Dette sprog har naturligvis sin egen syntaks etc. Programmerne må være udformet i dette sprog. Det sprog, vore programmer er udformet i, og som de datamater, vi bruger, kan "forstå", kaldes Algol 4 (den 4. version af sproget algol. Den første kom i 1960.). Det kan måske forvirre lidt, at vi også har kaldt Blok1 et sprog. Blok1 kan i denne forbindelse defineres som en særlig måde at skrive Algol-programmer på, et særligt dialekt om man vil, et dialekt der er tilpasset vore formål.

En beskrivelse af dette sprog findes i - A Manual of GIER Algol 4 - Sproget har den funktion, at det skal hjælpe operatoren (programøren) med at kommunikere med maskinen.

Evnen til at opfatte meddelelser i dette sprog findes som regel i maskinen, når man skal bruge den. For maskiner af type som GIER 1 på H. C. Ørstedinstituttet gælder der imidlertid, at maskinen kan glemme evnen. Man må da lære den det igen (evnen til at lære det har den dog stadig, jvf. næste afsnit). Det gøres på samme måde som når man indlæser et hvilket som helst program: sproget findes på et hulbånd mærket Ga4 (GIER Algol 4) og indputtes gennem læseren.

D. Om monitorsystemet HELP 3.

Evnen til at lære Algol 4 samt hele evnen til at kunne administrere sine egne funktioner og meddelelser fra og til operatoren findes også i maskinen. Disse evner findes i et såkaldt monitorsystem (overvågningssystem), der i dette tilfælde kaldes HELP 3 (den 3. version af monitorsystemet HELP). Dette system rummer også andre evner såsom evnen til at redigere og rette i indput (edit, se herom - Indføring i Blok1 II: Bloktrimning) og evnen til at huske **input** (reservere det) og glemme det igen (clear'e det), også det **meste** af sig selv og sit sprog.

De fleste af HELP 3's evner må opfattes som hjælpeprogrammer under HELP 3.

Systemet oversætter alle programmer skrevet i Algol 4 og desuden selve Algol-sproget (Ga4) til en repræsentation, der fylder

mindre i maskinen. Den siges at kompile dem (compile = samle, udarbejde register). For Algolprogrammernes vedkommende sker det i en række trin, de såkaldte passes. Er der sproglige fejl i programmet, meddeler HELP 3 dette med en angivelse af, i hvilken pass fejlen er fundet, og af hvilken art den er. Meddelelserne fra disse 9 passes kan slås op i - A Manual of GIER Algol 4 - side 80 - 84.

Som sagt kan maskinen også glemme det meste af HELP 3. det må da også indlæses. På GIER 1 sker det ved at indlæse båndet MAIN HELP (se - Indføring i Blok 1 II: Bloktrimning).

E. Om strukturen i en kørsel.

En kørsel med Blok 1 kan da siges at have følgende struktur gående inde fra maskinen og udad mod operatøren og samtidig gående i den tidsmæssige rækkefølge, hvori de enkelte faser forekommer.

1. HELP 3 aktiveres (indlæses evt.)
2. Algol 4 aktiveres (indlæses evt.)
3. Blokblokken indlæses og aktiveres.
4. Blokprogrammet indlæses og aktiveres.
5. Databeskrivelse indlæses, evt.defineres styringsparametre.
6. Data indlæses.
7. De egentlige operationer med data begynder her og slutter med
8. outputtet fra kørslen.

En mere konkret beskrivelse af, hvordan man kører med blok 1 gives i - Indføring i Blok 1 II: Bloktrimning - .

Litteratur.

- Abelson, R. P.: Simulation of Social Behavior. i Lindzey, G. og Aronson, E.: Handbook of Social Psychology. Addison-Wesley, Massachusetts 1968. Vol 2., s. 274 - 356.
- Dreier, O. : Udkast til en vejledning i brug af GIER med HELP 3. Duplikeret. Danmarks Lærerhøjskole 1967.
- - : **Indføring** i Blok II: Bloktrimning. SKETs monografiserie 9. Duplikeret. Danmarks Lærerhøjskole 1970.
- Karpatschoff, B. og Rattleff, J.: SORTERING. Duplikeret. Danmarks Lærerhøjskole. København 1967.
- Lauesen, S. et. al.: A. Manual of HELP 3. Regnecentralen. København 1967.
- Naur, P. : Plan for et kursus i Datalogi og Datamatik. Regnecentralen. København 1966.
- - et. al.: A Manual of GIER Algol 4. Regnecentralen. København 1967.
- Rattleff, J. : Datatrimning. Nogle overvejelser over datarepræsentation af psykologiske eksperimenter. SKETs monografiserie 4. Duplikeret. Danmarks Lærerhøjskole. København 1968.
- Rattleff, J. og de Waal, B. : Bloktrimning 2. Duplikeret. Danmarks Lærerhøjskole. København 1968.

MÅ IKKE NIEMLÅNES

