

# Interne notater

STATISTISK SENTRALBYRÅ

85/35

19. september 1985

PRAKTISK BRUK AV DEN GRAFISKE  
PAKKEN GIMMS

av

Geir-Harald Strand

## INNHold

	Side
Figurliste . . . . .	1
1. Innledning . . . . .	2
2. Filstrukturen omkring GIMMS . . . . .	3
3. JCL-kort for GIMMS . . . . .	7
4. Kommandospråk og syntaks . . . . .	11
5. Grunnleggende GIMMS-eksempel . . . . .	15
5.1. Polygonisering . . . . .	16
5.2. Tegning av omrisskart . . . . .	21
6. Fem kommenterte GIMMS-eksempler . . . . .	23
6.1. Geografisk grunnlag og omriss . . . . .	23
6.2. Systemfil med beskrivende data . . . . .	26
6.3. Koroplettkart med to histogrammer . . . . .	28
6.4. Sektordelte flateproposjonale sirkler . . . . .	33
6.5. Kurvedigram med 6 kurver . . . . .	38
7. Diverse forhold ved bruk av GIMMS . . . . .	42
7.1. Valg av tegnemaskin . . . . .	42
7.2. Kommandofilen . . . . .	43
7.3. Ferdige GIMMS-datasett . . . . .	44
7.4. Plassering av plottet på papiret . . . . .	45

FIGURLISTE

1. Simpel filstruktur i JCL-omgivelser . . . . .	3
2. GIMMS' filstruktur . . . . .	5
3. Grunnlagskart for koordinatfesting, eksempel . . . . .	16
4. Koordinatfesting av ett enkelt linjesegment . . . . .	18
5. Kommandoer for koordinatinput og polygonisering . . . . .	18
6. JCL-kort for koordinatinput og polygonisering . . . . .	20
7. Kommandoer for uttegning av enkelt polygonomriss . . . . .	21
8. JCL-kort for uttegning av enkelt polygonomriss . . . . .	21
9. Polygonomriss, linjeprintergrafikk generert med GIMMS . . . . .	22
10. Koroplettkart - GIMMS-eksempel nummer 3 . . . . .	31
11. Kart med kakediagram (pie-charts) - GIMMS-eksempel nr.4 . . . . .	37
12. Kurvediagram med 6 kurver - GIMMS-eksempel nummer 5 . . . . .	40
13. JCL-kort for allokering av nye datasett . . . . .	44
14. Hjørnekoordinater for ferdige polygondatasett . . . . .	45
15. Beregning av skaleringsfaktor for å sette målestokk . . . . .	46

## 1. INNLEDNING

Dette notatet gir en innføring i bruk av den grafiske programpakken GIMMS på SSB's IBM-anlegg. Ved siden av koblingen mellom program og operativsystem er det lagt vekt på å eksemplifisere bruken av pakken, for på den måten å gi en innfallsvinkel til kommandospråket. For å få en uttømmende oversikt over språk og syntaks må man imidlertid benytte GIMMS brukermanual.

Kapittel 2 gir en oversikt over de omgivelsene GIMMS fungerer i, med vekt på å forklare filstrukturen som er knyttet til bruken av programmet. Fremstillingen er skjematisk og skal fremfor alt fungere som et grunnlag for de neste kapitlene.

Kapittel 3 gjennomgår den JCL'en som er nødvendig for å kjøre GIMMS. Størsteparten av JCL'en er knyttet til allokering av datasett.

Kapittel 4 er en overfladisk innføring i kommandospråket og syntaksen som benyttes i GIMMS.

Kapittel 5 inneholder et eksempel på overføring av koordinatdata fra kart til GIMMS-systemet. Ved siden av den grunnleggende fremgangsmåten ved digitalisering tar kapitlet opp såvel kommandostruktur som JCL knyttet til dette arbeidet.

Kapittel 6 gjennomgår 5 eksempler på praktiske kjøringar som delvis er hentet fra GIMMS-manualen, delvis fra et sett med eksempler som ble levert sammen med programmet. Ved siden av eksemplene inneholder kapitlet en god del kommentarer som forklarer programoppsettene. Dette bør kunne fungere som en "lærebok" som gir en grunnleggende innføring i GIMMS' kommandoer og syntaks, og gjør det lettere for brukeren senere å orientere seg i GIMMS-manualen.

Kapittel 7 er et oppsamlingskapittel som beskriver endel særegenheter knyttet til bruken av GIMMS på IBM.

Det gjøres oppmerksom på at hva gjelder datamaskinen og systemet omkring denne, så inneholder teksten en rekke forenklete fremstillinger. Dette er tilsiktet. Notatets målgruppe er GIMMS-brukere på fagkontorene, og den nødvendige behandling av maskinnære forhold og bruk av fagterminologi er søkt tilrettelagt for å lette forståelse og bruk av systemet i denne gruppen.

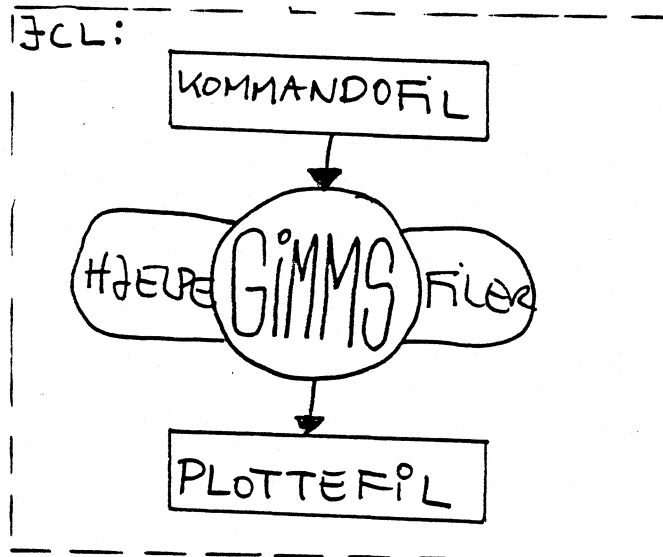
## 2. FILSTRUKTUREN OMKRING GIMMS

Dette kapittelet tar sikte på å gi en kortfattet oversikt over filstrukturen omkring den grafiske pakken GIMMS. Kapittelet gir ikke konkrete eksempel på innholdet av de forskjellige filene, men bør kunne tjene som et grunnlag for å forstå slike eksempler når de blir lagt frem i senere kapitler.

Figur 1 gir en oversikt over en simpel GIMMS-struktur. Selve GIMMS er et program som startes opp med en EXEC-kommando i en JCL-jobb. Filen med JCL-kortene må også inneholde de nødvendige kommandoer (DD-kort) for å allokere de filene / datasettene som GIMMS skal benytte. I figur 1 er disse vist ved hjelp av tre bokser:

### 1. KOMMANDOFIL

Kommandofilen inneholder beskjeder til GIMMS-programmet som beskriver det arbeidet programmet skal utføre. Disse kommandoene legges inn med en særlig syntaks, som er beskrevet i GIMMS-manualen. Å lage en kommandofil til GIMMS kan sammenlignes med å programmere i SAS eller EASYTRIEVE.



Figur 1: Simpel filstruktur for GIMMS i JCL-omgivelser.

Det må ALLTID være med en kommandofil når GIMMS skal kjøres.

## 2. PLOTTEFIL

Når GIMMS produserer grafikk sendes ikke denne direkte til tegne-maskinene, men lagres som instruksjonssymboler på en egen fil: Plottefilen. Denne filen kan siden sendes over til tegnemaskinen på samme måten som når en tekstfil sendes til en printer. Plottefil trenger vi bare ha med i de tilfellene hvor GIMMS faktisk benyttes til å tegne grafikk.

## 3. HJELPEFILER

I figur 1 er det lagt en egen samleboks kalt hjelpefiler under GIMMS-programmet. Dette er en samling på omkring 5 filer som alltid må være med.

De fleste GIMMS-jobber vil inneholde disse filene. Alle filene deklarerer ved hjelp av DD-kort i JCL-filen som omslutter jobben.

For å kunne tegne grafikk trenger GIMMS et datagrunnlag som skal fremstilles grafisk. Også disse dataene må på noen måte befinne seg som filer i maskinen. Videre skal de være bygget opp på et særskilt format, noe som gjør at vi ikke er istand til å punche disse filene. Data av ymse slag må ligge på GIMMS "systemfiler". Slike filer kan bare produseres av GIMMS selv. Dette er illustrert i figur 2.

I figur 2 finner man igjen elementene fra figur 1: Selve GIMMS-programmet samt kommandofilen, plottefilen og hjelpefilene. De nye elementene som er kommet inn skal beskrive gangen i oppbygging og bruk av systemfiler.

Input av data til GIMMS går via kommandofilen. Til venstre for denne er det tegnet på tre datatyper: Punkter, Segmenter og Data. Med punkter mener vi koordinatfestede punkter, gitt ved X- og Y-koordinat. Med segmenter menes linjer beskrevet ved en serie punkter, og "data" er i denne forbindelsen statistiske variabler som beskriver de geografiske enhetene (folketall, skoleplasser, skutte elg). Inne i en kommandofil kan vi legge inn et slikt datasett og få dette lest av GIMMS. GIMMS omdanner så dataene til systemfiler. Disse er tegnet som bokser til høyre for programmet i figuren. Siden kan GIMMS lese data tilbake fra systemfilene og benytte dem for å generere grafikk.

Systemfilene må også deklarereres med DD-kort i jobbens JCL. Når en GIMMS-kjøring leser inn data via kommandofilen og skriver dem ut på en systemfil, opprettes denne filen på DD-kortet. Her kan brukeren velge om filen skal opprettes som temporær (DISP=(NEW,DELETE,DELETE)) og altså slettes etter bruk, eller om den skal tas vare på (DISP=(NEW,KEEP,DELETE)). Tar man vare på en systemfil kan data hentes direkte fra denne inn i nye GIMMS-jobber - det er ikke nødvendig å entre data på nytt via kommandofilen.

#### 4. PUNKTFIL

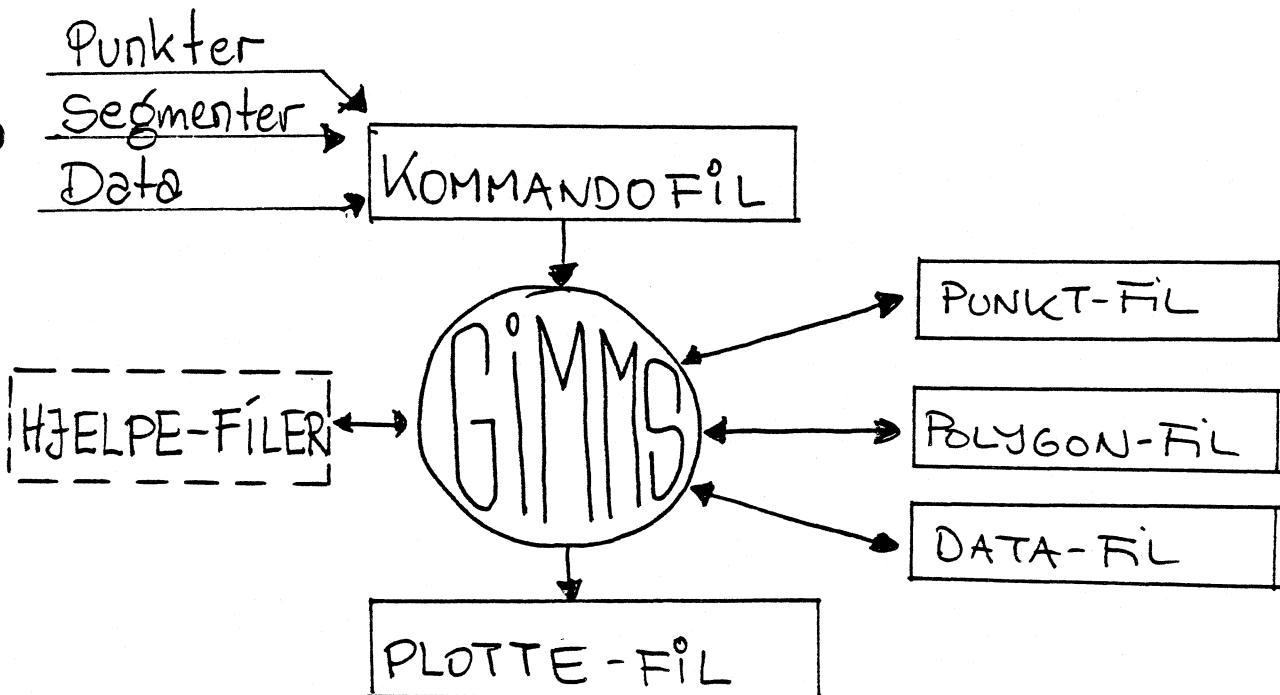
Punktfilen er en systemfil som inneholder definisjon av koordinat-festede punkter. Disse punktene kan vi benytte for å tegne inn punktsymboler på et kart.

#### 5. SEGMENTFIL

Segmentfilen inneholder linjestykker definert ved et sett koordinat-festede punkter. Segmentfilen kan benyttes til å tegne linjer på et kart, eller til å bygge opp polygoner (arealer) som det skal festes kartsymboler ved.

#### 6. POLYGONFIL

Polygonfilen har ikke noe korresponderende datainputtype i kommandofilen. For å bygge opp og lagre polygoner må vi først legge inn segmentdata i kommandofilen og overføre disse til en segmentfil.



Figur 2: GIMMS filstruktur. JCL kommer i tillegg.

Denne segmentfilen hentes så inn på nytt av GIMMS som bygger opp polygoner ved hjelp av segmentene, og de definerte polygonene lagres så på polygonfilen. Hele dette arbeidet kan gjøres i en enkelt GIMMS-jobb. Som oftest har vi ikke behov for å bevare systemfilen med segmentene, mens vi vil ta vare på polygonene. Segmentfilen defineres da som en temporær fil, mens polygonfilen bevares.

#### 7. DATAFIL

Datafilen er en systemfil som inneholder statistiske variabler. Den kan brukes til å tegne kurver, histogrammer og kakediagram. Om det finnes punkt- og/eller polygonfiler som beskriver den geografiske plasseringen av enhetene i denne filen kan datafilen også benyttes til utforming av kart.

### 3. JCL-KORT FOR GIMMS

#### JOB-kort

JOB-kortet for GIMMS-kjøringer fylles i hovedsak ut som vanlig. GIMMS bruker imidlertid mer plass - og iblant mer tid - enn hva som automatisk tildeles en jobb. REGION- og TIME-parameter for å rette på dette kan legges inn på JOB-kortet. I dette notatet er de imidlertid lagt på EXEC-kortet som starter GIMMS, noe som fungerer like bra.

#### EXEC-kort

Exec-kortet starter selve GIMMS-kjøringen. Det har formen:

```
//GIMMSTEP EXEC PGM=GIMMS,REGION=1600K,TIME=(2,0)
```

- PGM-parameteren henter frem GIMMS-programmet. Datasettet hvor dette programmet er lagret må allokeres på eget STEPLIB DD-kort (se under).
- REGION-parameteren setter av arbeidsplass for GIMMS. Denne må være med fordi GIMMS benytter mer plass enn det som vanligvis settes av til en programeksekvering. Parameteren kan også legges på JOB-kortet.
- TIME-parameteren forteller hvor lenge programeksekveringen kan pågå før den brytes. I eksemplet er dette 2 minutter. Ved store kart-jobber kan det være nødvendig å utvide tidsparameteren ytterligere. TIME-parameteren kan også legges på JOB-kortet.

#### DD-KORT

GIMMS krever allokering av en rekke filer. Filsystemet knyttet til GIMMS er skjematisk beskrevet i kapittel 2.

Kommandofilen for GIMMS som inneholder instruksjoner om utformingen av grafikken (arbeidsinstrukser for programeksekveringen) er en fil som brukeren selv redigerer ved hjelp av editor. Samtidig er det en fil som skal leses av en batch-jobb. Den må derfor befinne seg på en SSB-disk. En slik allokering er ikke mulig fra ISPF-menyen 3.2, og datasettet der kommandofilen ligger må følgelig allokeres i en egen batch-jobb. Fremgangsmåten for dette er beskrevet i kapittel 7.



Videre er det nødvendig med et eget DD-kort for å angi hvilket datasett selve GIMMS-programmet ligger i. DDnavnet her må være STEPLIB. Dette datasettet er ikke behandlet i kapittel 2 siden det ikke er en del av selve GIMMS-strukturen.

#### STEPLIB

STEPLIB DD angir hvor GIMMS-programmet ligger. Dette bør være det aller første JCL-kortet etter EXEC-kortet. Idag vil dette kortet ha utformingen:

```
//STEPLIB DD DSN=SSB1.GIMMS.LOAD,DISP=SHR
```

Datasettnavnet på dette kortet kan imidlertid komme til å bli endret idet det der mulig at programmet blir overført til et annet datasett. Dette kommer an på Driftskontorets disposisjoner - så følg med i meldingene derfra.

#### HJELPEFILER

7 hjelpefiler må alltid allokeres i en GIMMS-jobb. Følgende utforming kan følges slavisk:

```
//FT06F001 DD SYSOUT=*  
//FT09F001 DD DSN=SSB1.GIMMS.ALPHABET,DISP=SHR,DCB=BUFNO=1  
//FT17F001 DD UNIT=DISK,SPACE=(1000,1000),DCB=BUFNO=1  
//FT20F001 DD UNIT=DISK,SPACE=(TRK,(20,10)),  
//          DCB=(RECFM=VB,LRECL=1028,BLKSIZE=3088,BUFNO=1)  
//FT21F001 DD UNIT=DISK,SPACE=(1000,400),DCB=BUFNO=1  
//FT22F001 DD UNIT=DISK,SPACE=(1000,400),DCB=BUFNO=1  
//FT23F001 DD UNIT=DISK,SPACE=(1000,2000),DCB=BUFNO=1
```

DD-navnene her er faste på formen FTnnF001, og for hjelpefilenes del må de angitt nn-sifferene også følges. De syv nummerene over er dermed ikke tilgjengelige som interne filnummere i GIMMS. Enkelte andre nummere er også reserverte. Tilsammen 13 slike nummere finnes:

05, 06, 07, 09, 17, 18, 19, 20, 21, 22, 23, 24

Alle andre nummere mellom 00 og 99 er i prinsippet tilgjengelige. Det anbefales imidlertid å benytte nummere over 10 som filnummere i GIMMS. Systemfilene kan dermed ha nummere i størelsesorden 10-16 og 25-99.

Av de reserverte nummerene brukes endel til hjelpefilene som angitt over. 05 benyttes til kommandofilen og 19 til plottefilen. De øvrige er reservert for eventuell fremtidig utvidelse av GIMMS-pakken.

#### KOMMANDOFILEN

Kommandofilen må ligge på et datasett allokert på en SSB-disk. Den inneholder instruksjoner til GIMMS skrevet i et eget kommandospråk som er beskrevet i GIMMS-manualen. Kommandofilen allokeres til enhet 5.

```
//FT05F001 DD DSN=TK22.S6155.BATCH(KONG01),DISP=SHR
```

Alternativt kan kommandoene legges inn in-stream på samme fil som JCL'en. Det vil da ta seg slik ut:

```
//FT05F001 DD *  
gimmskommando.....  
:  
:  
gimmskommando .....  
/*
```

Man følger altså samme prosedure som ellers ved innlegging av data in-stream.

#### PLOTTEFILEN

Om GIMMS-jobben skal produsere grafikk mellomlagres denne som kommandoer (tegneinstruksjoner) på en egen plottefil som allokeres til enhet 19. Plottefilen sendes siden over til tegnemaskinen som man ellers sender en fil til en printer. DD-kortet for plottefilen har formen:

```
//FT19F001 DD DSN=TK22.S6155.PLOTTEFIL,DISP=(NEW,CATLG,DELETE),  
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=4080),  
//          UNIT=DISK,SPACE=(4080,(50,10),RLSE),VOL=SER=SSB002
```

Har man en eksisterende plottefil kan denne overskrives med DISP=OLD. De øvrige (etterfølgende) parametrene droppes da. Noe spesiell er plottefilen for linjeskriver (DEVICE=2). LRECL settes da til 132 og BLKSIZE til 4092. Glemmer man dette vil GIMMS-jobben havarere og man får melding om en Fortran I/O feil.

#### SYSTEMFILER

Systemfiler er filer der GIMMS lagrer data, koordinater og polygon-beskrivelser. Disse refereres det til fra kommandofilen ved hjelp av filnummere som må korrespondere med DD-navnet filene allokeres med. Refereres det til en fil med GIMMS-kommandoen FILE=10 må denne filen allokeres som FT10F001. Viktig i denne omgang er DCB-parameteren på formen DCB=(RECFM=VB,LRECL=1028,BLKSIZE=m) der m er et tall konstruert med  $m=n+1028+4$  og hvor n kan være et hvilket som helst heltall. Med

n = 3 blir m = 3088.

En systemfil som ikke skal bevares opprettes på formen:

```
//FT10F001 DD DSN=TK22.S6155.POLYGON,DISP=(,DELETE,DELETE),  
//          DCB=(RECFM=VB,LRECL=1028,BLKSIZE=1032),  
//          UNIT=DISK,SPACE=(1032,(10,10),RLSE)
```

Ved å forandre DISP-parameteren til DISP=(NEW,CATLG,DELETE) vil datasettet bli bevart etter at jobben er ferdig kjørt. Når et datasett bevares på denne måten kan det siden hentes inn igjen enkelt ved:

```
//FT10F001 DD DSN=TK22.S6155.POLYGON,DISP=SHR
```

Legg nå merke til at datasettnavnet i disse eksemplene må tilpasses av brukeren, og at DD-navnet FT10F001 kun er ett av en serie mulige og avhenger av filnummertilknytningen i kommandofilen.

#### KOMPLETT EKSEMPEL

For et komplett eksempel på en GIMMS-jobb med JCL henvises det til kapittel 5.

#### 4. KOMMANDOSPRÅK OG SYNTAKS

KOMMANDO: En ordre til GIMMS. Selve kommandoordet innledes med en stjerne (\*), eks: \*SYSPARM

SUBKOMMANDO: Tilleggsinformasjon som beskriver nærmere hvordan en kommando skal utføres. En subkommando kan være logisk, eks:

\*SYSPARM LOGO

eller den kan ta verdi. Verdien må da henges ved som PARAMETER:

\*SYSPARM DIAGS=2

En og samme kommando kan ta flere subkommandoer, og samme subkommando kan igjen ta mer enn en parameter. Såvel subkommandoer som parametre skilles ad ved et komma (,) eller en eller flere blanke ( ).

I visse tilfeller kan man sløyfe selve subkommandoen og bare skrive parameteren. Det er også lovlig å sløyfe tegnet '=' mellom subkommando og parameter. Dette er ofte gjort i den engelske GIMMS-manualens eksempler, og kan virke forvirrende.

KORT: Et kort er en kommando med subkommandoer og parametre. Et kort kan om nødvendig strekke seg over flere linjer.

MODUL: Logiske blokker av kommandoer. En GIMMS-jobb kan bestå av flere moduler, og modulene kan være nestet inn i hverandre. Modul er ikke noe strengt definert begrep i GIMMS, men snarere en måte å tenke på. En jobb som produserer en tegning kan vi tenke på som en stor modul. Denne inneholder en mindre modul for definering av systemparametre, en for å hente inn data og en for å utforme tegningen. Tegnemodulen kan igjen bestå av en data- og symbolmanipulerende modul, en karttegnemodul, en tegnforklaringsmodul og en tekstmodul.

KOMMANDOFIL: Et datasett der vi skriver inn alle kommandoer, subkommandoer og parametre til en enkelt GIMMS-jobb. Kommandofilen kan i tillegg inneholde endel data som skal benyttes i jobben.

```
*SYSPARM
*TEXTPARM
*PLOTPARM
  *FILEIN
  *POLYGON
  *SAVE
  *RESTORE
  *GIMMSFILE
  *FILEIN
  *MANIPULATE
    *COMPUTE
  *END
  *PLOTPROG
    *NEWMAP
    *INTERVALS
    *LEGEND
    *SYMBOLISM
    *SCALE
    *ORIGIN
    *DRAWMAP
    *MAP
  *END
  *GRAPHICS
    *NEWPAGE
    *SYMBOLISM
    *PEN
    *ORIGIN
    *GRAPH
    *BARGRAPH
    *PIE
    *SCATTER
  *END
  *TEXT
*STOP
```

Denne oversikten summerer opp de viktigste GIMMS-kommandoene, samtidig som innrykningen forsøker å vise hvordan enkelte kommandoer hører inn under moduler startet opp av andre kommandoer. Den eneste kommandoen som alltid er nødvendig er \*STOP, som avslutter ethvert GIMMS-program.

Plasseringen av kommandoene er imidlertid ikke på noen måte så rigid som dette skjemaet gir inntrykk av. Oppsettet er kun ment som en strukturert oversikt. Endel kommentarer er derfor nødvendige:

Om vi starter opp en modul med \*MANIPULATE, \*PLOTPROG eller \*GRAPHICS så må modulen avsluttes med \*END, noe som går frem av oversikten.

Kommandoene \*RESTORE, \*GIMMSFILE og \*TEXT kan også benyttes inne i en modul startet med \*PLOTPROG eller \*GRAPHICS.

Enkelte kommandoer er skrevet inn både under \*PLOTPROG og under \*GRAPHICS. Flere av kommandoene under disse to modulene kan imidlertid benyttes i begge moduler, selv om dette ikke går eksplisitt frem av skjemaet.

Eksemplene i kapittel 5 og 6 vil gi et innblikk i bruken av kommandoene, samt vise noen av de mulighetene som ligger i tilfanget av subkommandoer. For å få en fyldig beskrivelse av mulighetene i GIMMS er det imidlertid nødvendig å ty til manualen. Man må imidlertid da ta seg tid til å sette seg inn i måten denne er bygget opp på.

I GIMMS-manualen dekker hvert hovedkapittel (section) kommandoer for en spesiell type arbeid som logisk sett hører sammen. Ofte er dette kommandoer som hører inn under en spesiell modul. I hovedkapittelet er det så en rekke underkapittel. Hvert av disse tar for seg ett eller flere kommandoer. Det gis gjerne en kort forklaring av hva kommandoen benyttes til, deretter følger en liste over subkommandoer og til slutt beskrives hver enkelt subkommando i detalj. For å kunne bruke manualen effektivt må man først og fremst kunne tolke listene over subkommandoer, og vi skal derfor se nærmere på en slik. Her følger et utsnitt av listen over subkommandoer til \*BARGRAPH-kommandoen:

SAMEPAGE (NC)(NOCLEAR)	/K/	Produce graph on the current page
XMIN (START)	/R/	Start value of x axis
XMAX	/R/	End value of x axis
NOAXES (NOAXIS)	/K/	Do not draw axis or labels
LINEGRP	/I,REPEAT=20/	List of line types for each variable

Selve subkommandoen er skrevet med store bokstaver lengst til venstre. Iblant følger også ett eller flere ord i parentes. Dette er ord som kan brukes alternativt til den egentlige subkommandoen - ord som altså betyr det samme. Angivelsen mellom to "slasher" (/--/) midt på siden angir hva slags type data subkommandoen eventuelt tar, og teksten til

høyre gir en kortfattet beskrivelse av hva subkommandoen gjør. En liste med forklaring av de mest brukte type-angivelsene følger:

- /I/ - Heltall (tall uten komma).
- /R/ - Decimaltall - dvs. tall som kan inneholde komma.
- /S/ - Tekststreng. Denne må stå i gacketegn, eks: 'Her er en streng'
- /K/ - Subkommandoen er logisk og tar ikke verdi

Type-angivelser angitt med kolon (eks :P) kan man se bort ifra. De har ingen relevans ved batch-kjøring.

Typeangivelser fulgt av REPEAT=tall (eks: /I,REPEAT=20/) betyr at subkommandoen kan ta en rekke verdiparametre, maksimalt så mange som angitt ved "tall".

Iblant benyttes også = for å angi hvilken verdi GIMMS benytter om subkommandoen ikke benyttes for å legge inn noen parameter fra brukeren. For eksempel vil /I,=1/ bety at subkommandoen tar et heltall som verdi, men hviss brukeren ikke oppgir subkommandoen overhode vil det bli oppfattet som om den var gitt og hadde fått verdien 1.

Systemet for konstruksjon av slike subkommandolister er beskrevet i detalj på side 2.10 - 2.13 i GIMMS-manualen.

## 5. GRUNNLEGGENDE GIMMS-EKSEMPEL

Figur 2 (side 5) skisserer hovedtrekkene i filstrukturen omkring GIMMS. I tillegg til de faste filene benytter systemet en serie datafiler. Antallet og arten av slike vil variere fra jobb til jobb. Disse brukerfilene vil inneholde såvel geografiske som statistiske data som går inn som et grunnlag for tegnearbeidet. Brukerfilene kan vi klassifisere på to måter: For det første etter hva de inneholder, og for det andre etter måten de er organisert på. En grov innholdsklassifikasjon går mellom koordinatfiler som beskriver ulike typer geografisk stedfesting, og "datafiler" i snever forstand som inneholder statistiske (beskrivende) data.

Alle filer med brukerdata kan videre foreligge på to forskjellige måter. Den første, og mest familiære, er som formattede datasett. Dette er datasett slik de foreligger fra "punch'en", skrevet med bokstaver og tegn som kan leses av mennesker. Formattede datasett kan vi lese ved hjelp av datamaskinens EDIT- og BROWSE-faciliteter. Også GIMMS er istand til å lese slike filer.

Formattede filer er imidlertid ikke videre effektive for maskinen å arbeide med. GIMMS overfører derfor brukerdatasettene til en ny organisasjonsform - gjerne kallt "systemfiler" eller "maskinkodefiler" - før programmet begynner å arbeide med dataene. Slike systemfiler kan vi ikke lese ved hjelp av EDIT eller BROWSE. Når data leses inn fra formattede filer, må disse allokere (ved hjelp av DD-kort) og gis parameteren DISP=OLD. Er det geografiske data som leses inn må det også alltid allokere korresponderende datasett for maskinkoden. For statistiske data er dette kun nødvendig om de skal bevares etter kjøring.

Om et datasett i maskinkode-format blir bevart etter en jobb, så kan dette datasettet benyttes direkte ved nye kjøring. Denne fremgangsmåten er tids- og ressursbesparende og bør i størst mulig grad benyttes. Legg forøvrig merke til at brukerdatasettene ikke trenger ligge på egne, separate filer, men kan legges inn som en del av kommando-filen til GIMMS-jobben om en finner dette enklere.

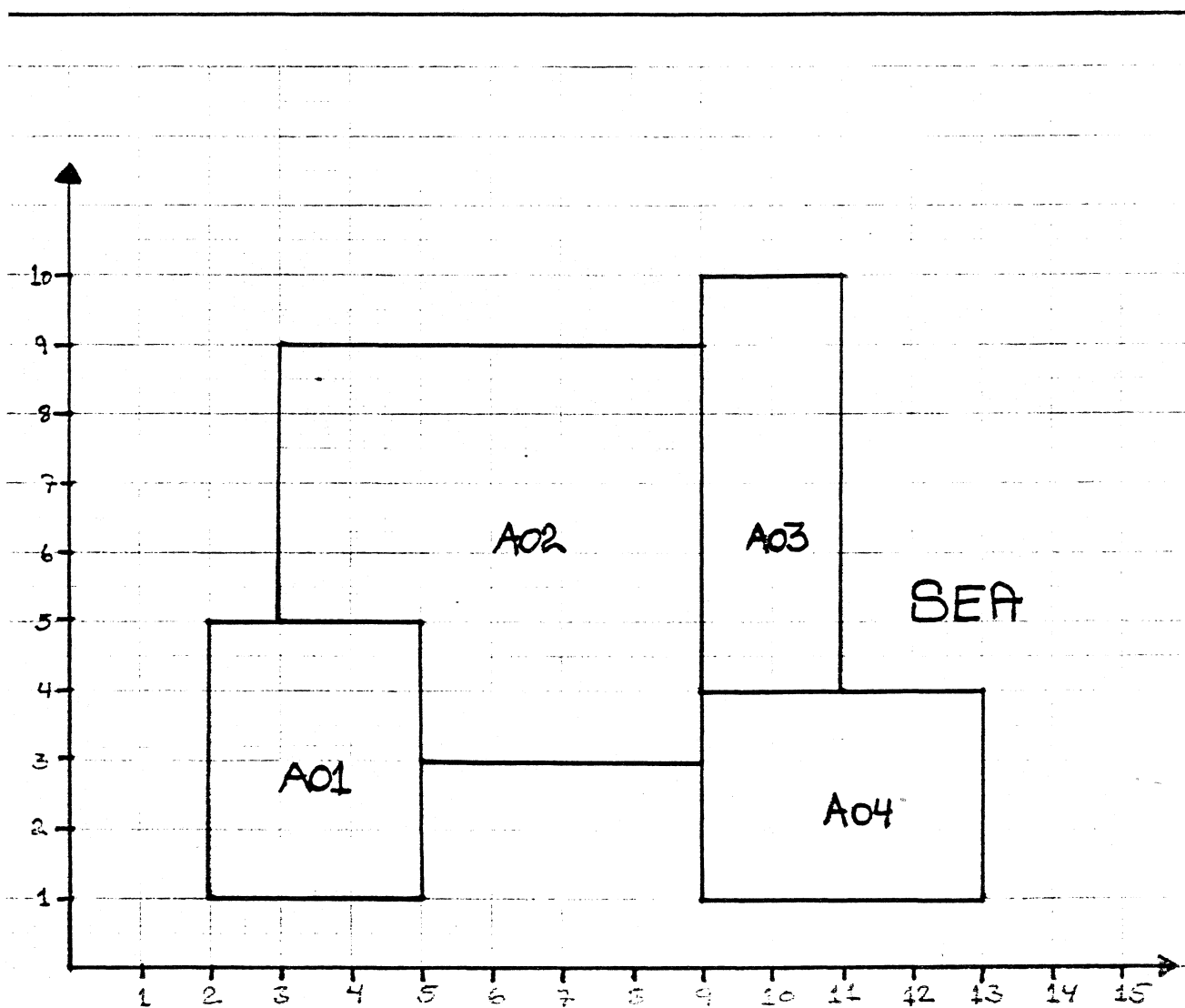
Resten av dette kapittelet vil ta for seg innlegging av koordinater for et enkelt "kart" og videre uttegning av dette. Alle kommandoer og komplett JCL for jobben følger eksemplet.



### 5.1 POLYGONISERING

Utgangspunktet for eksemplet er et tenkt kart bestående av fire arealer som skal koordinatfestes og tilrettelegges for GIMMS.

I figur 3 er kartet plassert inn i et koordinatsystem. Videre er alle arealene gitt enkle navn som innledes med en bokstav (A01, A02...). Til slutt er "tomrommet" utenfor arealene også gitt et navn, i dette tilfellet "SEA" (eng: Sjø). Grensene omkring arealene er trukket opp med en serie rette streker. Vi kunne benyttet flere streker og fått med mer detaljer, men i eksempeløyemed er kartet søkt gjort så enkelt som mulig.



Figur 3: "Kart" med nummererte områder plassert i et koordinatsystem for overføring til GIMMS.

Grunnlaget for kartkonstruksjonen vil nå være en koordinatfesting av enkeltpunkter. De punktene som skal koordinatfestes er

- 1/ Alle punkter hvor en linje skifter retning
- 2/ Alle punkter hvor to eller flere linjer møtes.

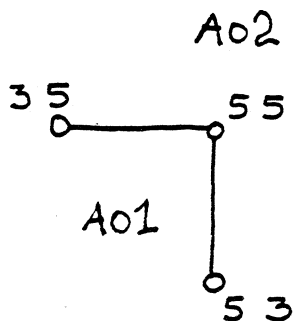
Koordinatfestingen skjer ved en tallfesting av punktet der første tall angir punktets X-koordinat og andre tall angir punktets Y-koordinat. Merk at det her er snakk om tall, som altså kan bestå av flere siffer. Tallene skilles ad med åpenrom eller komma. Således vil det nederst, venstre hjørnet i arealet A01 ha koordinat 2 1 mens det øverste høyre hjørnet i areal A04 har koordinat 13 4.

Neste trinn er å skille ut linjesegmenter i figuren. Et linjesegment består av en eller flere rette linjer mellom to punkter hvor flere linjer møtes. Et linjesegment får videre ikke inneholde noen punkter hvor flere linjer møtes. Ser vi nærmere på grensen omkring A01 består denne av to linjesegmenter. Ett utgjør grensen mot A02, det andre er grensen mot SEA. Grensen omkring A02 er mer komplisert, den består av 5 linjesegmenter. Grenselinjene mot de tre andre arealene utgjør ett linjesegment hver. Dessuten har A02 to linjesegmenter som grenser mot SEA, tilsammen 5.

Det er linjesegmentene som er input til GIMMS. Koordinatene angis for hvert enkelt punkt langs linjen fra det første knutepunktet (møtet mellom to linjer) til det andre knutepunktet. Linjesegmentene angis sammen med opplysning om hvilke arealer som befinner seg på hver side av linjen. Vi må derfor avgjøre hvilket knutepunkt vi vil starte koordinatangivelsene med, og hvor vi skal ende. Når dette er bestemt kan vi tenke oss at vi beveger oss langs linjen fra det første knutepunktet mot det andre. Navnet på arealet som da ligger til høyre skal alltid angis først, arealet til venstre deretter.

Figur 4 er et utsnitt av figur 3, hvor kun grenselinjen mellom arealene A01 og A02 er angitt. Linjesegmentet består av tre koordinatfestede punkter, hvorav to er knutepunkter. Knutepunktene har henholdsvis koordinatene 3 5 og 5 3. Mellom disse ligger punktet 5 5. Vi velger nå å ta utgangspunkt i knutepunktet 5 3 og bevege oss mot 3 5. Rekkefølgen på koordinatene vi legger inn blir da: 5 3, 5 5 og 3 5. Under denne bevegelsen ligger arealet A02 til høyre og skal nevnes først, mens A01 ligger til venstre og skal nevnes som nummer to. Hele linjesegmentet legges da inn med:

A02 A01 5 3 5 5 3 5/



Figur 4: Ekstrakt av figur 3 (s. 16) som viser koordinatfesting av ett enkelt linjesegment. Forklaring i teksten.

På denne måten legges alle linjesegmenter inn én gang. Også de segmentene som grenser mot ytterpolygonet (SEA i eksemplet) må legges inn. Tegnet "slash" (/) legges inn som separator etter hvert linjesegment. Man må være nøye med å få med alle punktene langs et linjesegment, og med å være konsistent i bruken av knutepunktkoordinater. To linjesegmenter som ender i samme knutepunkt må ha nøyaktig samme angivelse av koordinatene for dette knutepunktet.

Figurene 5 og 6 inneholder komplette kommando- og JCL-filer for eksemplet. Koordinatdataene er her lagt inn som en del av kommandofilen.

```
#SYSPARM,DIAGS=1,NOLIST,BATCH
#FILE IN
  FILEOUT=10,FILENAME=TESTSEG
  TYPE=SEGMENT
  LIMITS=0.0,15.10
  TITLE='FIKTIVE SEGMENTER'
  BEGIN
    A01 SEA 5 3 5 1 2 1 2 5 3 5 /
    A01 A02 3 5 5 5 5 3 /
    SEA A02 5 3 9 3 /
    A02 SEA 3 5 3 9 9 9 /
    A02 A03 9 9 9 4 /
    A02 A04 9 4 9 3 /
    A03 SEA 9 9 9 10 11 10 11 4 /
    A03 A04 11 4 9 4 /
    A04 SEA 11 4 13 4 13 1 9 1 9 3 /
  END *
#POLYGON FILEIN=10,FILEOUT=11,FILENAME=TESTSEG
  TITLE='FIKTIVE POLYGONER'
  MINZONE=A01,MAXZONE=A04
  ALPHA
*STOP .
```

Figur 5:

GIMMS-kommandoer for koordinatinput og polygonisering av et enkelt områdebasert kart.

SYSPARM-kommandoen setter systemparametre for jobben: Viktigst her er subkommandoen BATCH som angir at jobben kjøres i BATCH.

FILEIN-kommandoen utfører selve innlesningen av dataene. Fileout=10 angir her et nummer på en systemfil hvor segmentene skal skrives ut igjen. Siden vi velger å benytte filnummeret 10, så må denne filen i jobbens JCL allokere til FT10F001. FILENAME=TESTSEG gir denne filen et internt navn. Dette er ikke nødvendig, men kan være lurt i referansehensyn. Navnet blir lagret på systemfilen som dannes, men kan bare benyttes internt i GIMMS. Subkommandoen to linjer lenger nede, TITLE=' ..... ' lager likeledes en lengre tittel som også lagres på filen. Om filen blir tatt vare på kan vi siden undersøke hva som ligger på den med en \*FILEINFO-kommando, og vil da få listet ut både filename og title om slike er lagt inn. Legg merke til at tittelen skrives i apostrof.

Under FILEIN-kommandoen angir vi også at TYPE=SEGMENT hvilket betyr at de dataene som nå skal leses inn er linjesegmenter. Til sist beskriver vi et rektangel som kartet befinner seg inne i ved hjelp av subkommandoen LIMITS. Her følger koordinatene for nedre venstre (0 0) og øvre høyre (15 10) hjørne i et rektangulært kartutsnitt som omslutter kartet. Denne avgrensningen brukes senere av GIMMS for automatisk plassering av tegningen på papiret.

Selve innleggingen av linjesegmentene startes med subkommandoen BEGIN og avsluttes med END. Metoden som benyttes for å kode de enkelte segmentene er som skissert tidligere i dette kapitlet.

Omformingen av linjesegmenter til polygondefinisjoner skjer ved hjelp av kommandoen POLYGON. Grunnlaget for dette arbeidet er de innleste linjesegmentene som leses fra fil 10, og resultatet legges ut igjen på fil 11 (FILEIN=10,FILEOUT=11). Også her benyttes mulighetene til å legge et internt navn og en tittel inn på den nye filen.

I koordinatgrunnlaget for jobben har vi definert de fire arealene A01, A02, A03 og A04. I tillegg benytter vi arealet SEA for å beskrive "restområdet" i ytterkantene av kartet. Dette siste området vil vi ikke ha polygonisert - kun de fire "virkelige" arealene. Dette angis med subkommandoene MINZONE=A01,MAXZONE=A04 og ALPHA. ALPHA betyr i denne sammenheng at alle områder som er nevnt i segmentfilen skal sorteres alfabetisk. Fra denne alfabetiske listen er det så vi trekker ut arealene fra og med A01 til og med A04, konstruerer polygonene og legger disse ut igjen på fil nummer 11.

Den avsluttende STOP-kommandoen avslutter hele GIMMS-kjøringen.

Med hensyn på JCL'en for denne jobben legger vi i første omgang merke til at det ikke skal konstrueres noen tegning. Det er derfor ikke nødvendig å allokere FT19F001. Vi har likevel to brukerdatasett samt kommandofilen som må allokeres i tillegg til de faste datasettene. Kommandofilen kan vi tenke oss befinner seg på datasettet:

TK22.S6270.BATCH(SEGINN) som må allokeres til FT05F001.

Selve koordinatgrunnlaget befinner seg også på dette datasettet, så det volder ingen bekymring. Imidlertid må det konstrueres systemfiler for såvel segment-data som polygondata. Disse skal ligge på fil nummer 10 og 11 og må allokeres til FT10F001 og FT11F001. Segmentfilen vil bli kalt TK22.S6270.SEGMENT, mens polygonfilen skal få navnet TK22.S6270.POLYGON. I og med at arealene polygoniseres omgående er det ingen grunn til å ta vare på segmentfilen. Følgelig allokeres denne slik at den slettes straks jobben er slutt. Polygonfilen blir derimot bevart for senere bruk. Det komplette settet av JCL-kort er gjengitt i figur 6.

```
//LK22GHSA JOB (6270),'GEIP H STRAND',MSGCLASS=X,NOTIFY=LK22GHS,
//          MSGLEVEL=(0,0),CLASS=A,REGION=1600K,TIME=(2,0)
/*ROUTE PRINT RMIS
//GIMMS EXEC PGM=GIMMS,REGION=1600K,TIME=(2,0)
//STEPLIB DD DSN=SSB1.GIMMS.LOAD,DISP=SHR
//FT05F001 DD DSN=TK22.S6270.BATCH(SEGINN),DISP=SHR
//FT10F001 DD DSN=TK22.S6270.SEGMENT,DISP=(NEW,DELETE,DELETE),
//          DCB=(RECFM=VB,RECL=1028,BLKSIZE=1032),
//          UNIT=DISK,SPACE=(1032,(10,10),RLSE)
//FT11F001 DD DSN=TK22.S6270.POLYGON,DISP=(NEW,CATLG,DELETE),
//          DCB=(RECFM=VB,RECL=1028,BLKSIZE=1032),
//          UNIT=DISK,SPACE=(1032,(10,10),RLSE)
//FT06F001 DD SYSOUT=*
//FT07F001 DD *
//FT09F001 DD DSN=SSB1.GIMMS.ALPHABET,DISP=SHR,DCB=BUFNO=1
//FT17F001 DD UNIT=DISK,SPACE=(1000,1000),DCB=BUFNO=1
//FT20F001 DD UNIT=DISK,SPACE=(TRK,(20,10)),
//          DCB=(RECFM=VB,RECL=1028,BLKSIZE=3088,BUFNO=1)
//FT21F001 DD UNIT=DISK,SPACE=(1000,400),DCB=BUFNO=1
//FT22F001 DD UNIT=DISK,SPACE=(1000,400),DCB=BUFNO=1
//FT23F001 DD UNIT=DISK,SPACE=(1000,2000),DCB=BUFNO=1
```

Figur 6: JCL for kjøring av GIMMS' polygoniseringsjobb fra figur 5. Oppsettet er kommentert og forklart i teksten ovenfor.

Neste eksempel benytter den lagrede polygonfilen for å tegne ut et simpelt kart. Oppsettet for kommandofilen er vist i figur 7. Kommandoene SYSPARM og STOP har samme funksjon som tidligere. Ved tegning må vi videre benytte kommandoen PLOTPARM for å sette overordnede parametre omkring denne delen av jobben. Her benyttes subkommandoen DEVICE for å velge tegnemaskin. Tegnemaskin nummer 2 er linjeskriver.

```
*SYSPARM  DIAGS=1  NOLIST  BATCH
*PLOTPARM  DEVICE  =1
*DRAWMAP  FILE=11  OUTLINES=ALL  NAMES  MAPSIZE=25,18
*STOP
```

Figur 7: Kommandofil for uttegning av enkelt polygonomriss.

Kommandoen DRAWMAP tegner rene kartomriss - den benyttes kun til dette og kan ikke fylle kartet med tematisk innhold. Her hentes datagrunnlaget fra fil 11 (FILE=11) hvor den eksisterende polygonfilen må være allokert. Vi angir videre at omrisset på alle polygonene skal trekkes opp og at det skal skrives på navn. Til sist angir vi at kartet skal tegnes inne i et område på 25 x 18 centimeter på papiret.

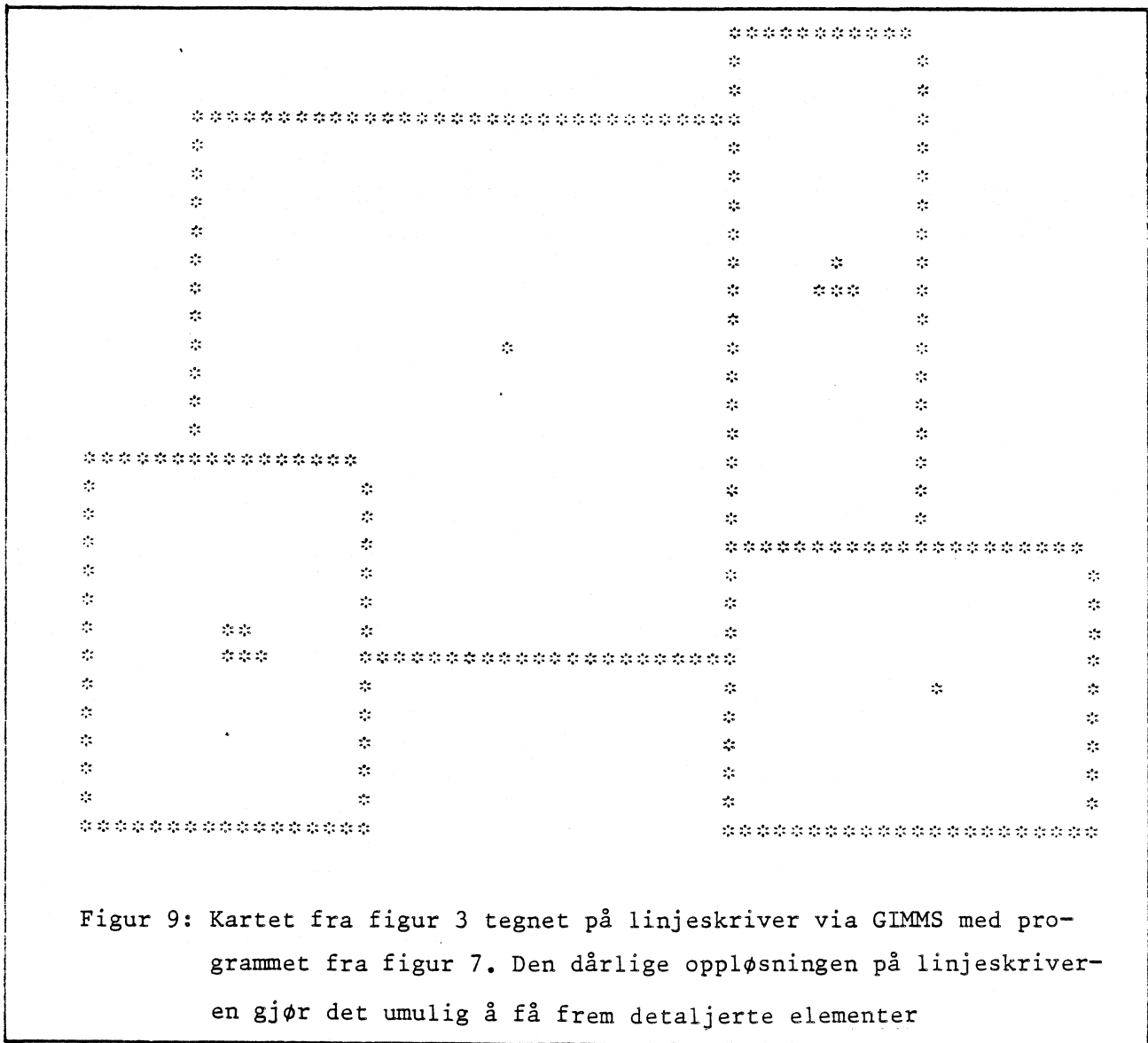
Den nødvendige JCL for å kjøre denne jobben er gjengitt i figur 8 nedenfor. Her opprettes ingen nye datasett ut over hjelpefilene. Kommando-

```
//LK22GHS  JOB (6270),*GEIR + STRAND*,MSGCLASS=X,NOTIFY=LK22GHS,
//          MSGLEVEL=(0,0),CLASS=A,REGION=1600K,TIME=(1,0)
/*JOBPARM  _INECT=70
/*ROUTE   PRINT RMTS
//GIMMS    EXEC PGM=GIMMS,REGION=1600K,TIME=(1,0)
//STEPLIB DD DSN=SSB1.GIMMS.LOAD,DISP=SHR
//FT05F001 DD DSN=TK22.S6270.BATCH(PL0TTJOB),DISP=SHR
//FT11F001 DD DSN=TK22.S6270.POLYGON,DISP=SHR
//FT19F001 DD SYSOUT=*
//FT06F001 DD SYSOUT=*
//FT07F001 DD *
//FT09F001 DD DSN=SSB1.GIMMS.ALPHABET,DISP=SHR,DCB=BUFNO=1
//FT17F001 DD UNIT=DISK,SPACE=(1000,1000),DCB=BUFNO=1
//FT20F001 DD UNIT=DISK,SPACE=(TRK,(20,10)),
//          DCB=(RECFM=VB,LRECL=1028,BLKSIZE=3088,BUFNO=1)
//FT21F001 DD UNIT=DISK,SPACE=(1000,400),DCB=BUFNO=1
//FT22F001 DD UNIT=DISK,SPACE=(1000,400),DCB=BUFNO=1
//FT23F001 DD UNIT=DISK,SPACE=(1000,2000),DCB=BUFNO=1
//
```

Figur 8: Nødvendig JCL for å kjøre GIMMS' jobben fra figur 7. Strukturen bak begge oppsettene er forklart i teksten.

filen ligger i dette tilfellet på TK22.S6270.BATCH(PLOTTJOB) som allokeres til FT05F001. Siden tegningen skal genereres for linjeskriver legges den ut i jobbets SYSOUT-strøm. Dette gjenspeiles ved allokering av FT19F001 til SYSOUT. Et poeng her er at JES2-kommandoen /~~M~~JOBPARM benyttes til å flytte sideskiftene på loggen slik at vi skal kunne få hele kartet inn på en enkelt side uten skip. Av brukerdatasett må den eksisterende polygonfilen fra forrige jobb allokeres til det filnummeret det skal leses inn via, her FT11F001.

Resultatet av jobben er gjengitt i figur 9, under. Ved å sammenligne med figur 3 ser man at det er samme kart. Linjeskriver er imidlertid et svært grovt tegneinstrument. Linjene er her risset opp med tegnet '\*'. Dette går på de enkle, rettlinjede grenseomrissene, men når man skal forsøke å tegne bokstaver (som også risses med rette streker) inne i arealene kan vi se at alt ender som en eneste klump i midten av hvert polygon. Prinsippene for jobben skulle imidlertid være illustrert.



Figur 9: Kartet fra figur 3 tegnet på linjeskriver via GIMMS med programmet fra figur 7. Den dårlige oppløsningen på linjeskriveren gjør det umulig å få frem detaljerte elementer

## 6. FEM KOMMENTERTE GIMMS-EKSEMPLER

Sammen med GIMMS-pakken ble det levert 5 filer med eksempler på GIMMS-programmer. Disse er lagret i datasettet SSBl.GIMMS.FORT på membrene DEMRUN01 til DEMRUN05. Eksempelene presenteres med kommentarer her - sammen med fotokopier av resultatene der slike forefinnes. Kjøringene dekker kart og annen grafikk omkring sosio-økonomiske forhold på Prince Edward Island - en øy ute i Gulf of St.Lawrence på syd-øst kysten av Canada.

### 6.1. Geografisk grunnlag og omriss

Første eksempel henter inn koordinater for linjesegmenter og skaper en GIMMS "segmentfil" hvor disse lagres. Deretter skapes en polygonfil på bakgrunn av segmentene og til sist tegnes det ut et omrisskart over øya hvor navnene på de ulike polygonene skrives på. Øya består av 29 områder som benevnes CA01, CA02 osv. til og med CA29.

```
*SYSPARM DIAGS=2,BATCH
```

```
*PLOT Parm DEVICE=4
```

```
*FILEIN SEGMENT FILEOUT=10 FILENAME=PEIBSF
```

```
TITLE='PRINCE EDWARD ISLAND SEGMENT FILE'
```

```
LIMITS=55500,47500,73850,55800
```

```
AUTONODE
```

```
BEGIN
```

```
CA01 OUT
```

```
57608 53643 57582 53664 57566 53674 57556 53685 57539 53695
```

```
:
```

```
her følger ca. 1500 linjer med koordinater
```

```
:
```

```
57679 52133 57674 52145 57674 52164
```

```
/
```

```
END
```

```
*POLYGON FILEIN=10 FILEOUT=11 FILENAME=PEIPOL
```

```
TITLE='PRINCE EDWARD ISLAND POLYGON FILE'
```

```
MINZONE=CA01,MAXZONE=CA29,ALPHA
```



```
GENERAL=800,600,400,200,100,50,10,5 REDUCE=1
*DRAWMAP FILE=11,OUTLINES=ALL,NAMES MAPSIZE=25,18
*FILEINFO FILE=11 ALL
*STOP
```

SYSPARM-kommandoen benyttes for å endre på systemparametre. To slike er satt i eksemplet: DIAGS og BATCH. DIAGS angir jobbens diagnosnivå - dvs. i hvor stor grad GIMMS skal legge ut rapporter om arbeidet på jobbens logg-fil. Færrest slike tilbakemeldinger får man med DIAGS=0, flest med DIAGS=4. Subkommandoen BATCH tar ikke parametre, men slår fast at jobben vil bli kjørt i batch-modus (submittet). Slik GIMMS nå er innstallert i SSB vil alle jobber bli kjørt i batch og denne subkommandoen kan derfor alltid være med.

PLOTPARM-kommandoen setter parametre omkring uttegningsarbeidet som skal skje. I eksemplet er den brukt til å angi hvilken tegnemaskin det skal lages grafisk kode for. I dette tilfellet maskintype 4 (Tektronix 4663 flatbed plotter). I SSBs installasjon av GIMMS må maskintype for uttegning alltid angis på denne måten - ved DEVICE=nummer under PLOTPARM-kommandoen. Den grafiske koden vil nå bli generert på fil 19 (FT19F001) som må allokeres til en fysisk fil ved hjelp av et DD-kort i jobbens JCL.

FILEIN-kommandoen leser inn data gitt på "menneskelesbar form", i dette tilfellet koordinatene for linjesegmenter. Koordinatene er punchet lenger nede på kommandofilen. Under FILEIN-kommandoen er det gitt en rekke subkommandoer: SEGMENT forteller at dette er en segment-beskrivelse - altså at dataene som leses inn er koordinater for linjesegmenter. FILEOUT=10 sier at segmentene når de er lest inn skal legges ut på en GIMMS' systemfil. Denne filen er i jobbens JCL allokert til FT10F001. FILENAME=PEIBSF gir systemfilen et eget referansenavn internt i GIMMS. TITLE gir mulighet til å legge et lengre navn inn på filen. Disse navnene vil bli skrevet ut om vi senere ber om opplysninger om hva som befinner seg på filene. LIMITS setter kartets yttergrenser i originalkoordinater. Disse beskriver et rektangel omkring selve kartet som benyttes når kartet skal passes inn på skjerm eller papir. Det er ikke nødvendig å angi nøyaktige ytterkoordinater, man bør isteden la det være en rimlig kant omkring selve kartet. LIMITS-koordinatene angis i følgende rekke: Vest, Syd, Øst, Nord (Lav X, Lav Y, Høy X, Høy Y). AUTONODE ber programmet selv beregne noder for eventuelle polygoner (midtpunkter) og BEGIN slår fast at nå begynner selve koordinatinnleggingen. Deretter følger ca. 1500 linjer med koordinater lagt inn på vanlig måte. Andre notater om GIMMS fra 12.kontor inneholder komplette eksempler på dette området. Koordinatinnleggingen avsluttes med stikkordet END.

På dette stadiet i jobben er segmentbeskrivelsene lest inn, konvertert til GIMMS-format og lagret som GIMMS systemfil på enhet 10.

POLYGON-kommandoen gir beskjed om å skape polygonbeskrivelser på basis av en segmentfil. FILEIN=10 forteller at segmentfilen ligger på enhet 10, FILEOUT=11 at polygonbeskrivelsene skal legges ut på enhet 11. Den fysiske filen hvor GIMMS skal plassere segmentbeskrivelsene er altså allokeret til FT10F001, og tilsvarende for polygonbeskrivelsene allokeret til FT11F001. Legg merke til at om de polygonbeskrivelsene som genereres i denne jobben skal kunne benyttes direkte i andre jobber senere, må DISP-parameteren i DD-kortet katalogisere og bevare datasettet. FILENAME=PEIPOL legger et GIMMS-navn på filen og TITLE hefter en lengre beskrivelse ved den. Som for segmentfilens vedkommende er dette navn som vi vil få listet ut om vi ber om en beskrivelse av filen. MINZONE=CA01 og MAXZONE=CA29 forteller hvilke av områdene som finnes i segmentfilen som skal polygoniseres. ALPHA legger til at denne utvelgelsen skal skje på alfabetiske kriterier. I eksempelet fører dette til at området utenfor øya - som i koordinatinputen ble kalt OUT - ikke blir forsøkt polygonisert. Man kan også tenke seg en situasjon der segmentfilen inneholder grensebeskrivelser for områder på ulike geografiske nivåer. For eksempel kommuner og fylker. Om kommunene var nummerert K0101 til K2030 og fylkene nummerert F01 til F20, kunne vi be om å få polygonisert kommunene ved å sette: MINZONE=K0101 og MAXZONE=K2030.

Subkommandoen GENERAL under POLYGON-kommandoen gir mulighet til å definere inntil 9 forskjellige generaliserings- (eller forenkling-) nivåer ved polygonene. I eksemplet angis 8 slike nivåer og de vil ha nummerene 1 til 8. Disse generaliseringsnivåene kan benyttes under uttegning av kartet (DRAWMAP) Ved generalisering trekkes en tenkt linje gjennom segmentets endepunkter og det beregnes hvor langt fra denne trend-linjen de øvrige punktene ligger. Det punktet som ligger lengst vekk benyttes som test-punkt. Ligger dette lenger vekk fra trend-linjen enn angitt som parameter under GENERAL for vedkommende generaliserings-nivå (ex: 400 enheter for generaliseringsnivå 3 i eksemplet) blir punktet ansett som signifikant. Linjen deles nå i to ved dette punktet og prosessen gjentas separat for de to halvdelene. Prosessen gjentas til det ikke lenger finnes signifikante punkter, og segmentet vil nå være redusert. (I virkeligheten er det hele noe mer komplisert - men forklaringen skulle gi en viss forståelse av prosessen). Ved uttegning av kart i svært liten målestokk får man på denne måten fjernet detaljer som ellers bare ville virke forstyrrende. REDUCE=1 angir en generaliserings-størelse for selve polygoniseringen. Denne er imidlertid gitt i kartkoordinater, ikke i generaliseringsnivåer. Generaliseringsnivåene gitt under GENERAL= kan hentes frem i DRAWMAP-kommandoen med GENERAL=nivå. For eksempel vil GENERAL=4 da bety at man benytter

generaliseringsnivå 4 som i eksemplet er satt som signifikante punkter 200 eller fler kartenheter fra trend-linjen.

DRAWMAP-kommandoen utfører en tegning. FILE=11 sier at kartgrunnlaget hentes fra enhet 11 (polygonfilen). OUTLINES=ALL ber om at alle grenser skal tegnes ut og NAMES at navnene på områdene skal skrives ut på kartet. MAPSIZE=25,18 er en beskjed om at kartet skal tegnes ut i et rektangel 25 cm. horisontalt og 18 cm. vertikalt på papiret. Under PLOTPARM-kommandoen ovenfor ble det bestemt av tegningen skal genereres for maskintype 4 og den grafiske koden for denne blir nå lagt ut på filenhet 19.

FILEINFO-kommandoen gir en beskrivelse av innholdet på en GIMMS systemfil. Beskrivelsen legges ut på jobbens logg-fil. FILE=11 angir at det er filen som er allokert til enhet 11 som skal beskrives og ALL at vi ønsker all tilgjengelig informasjon om denne.

STOP-kommandoen avslutter en GIMMS-jobb

Det finnes ikke noe uttegnert eksempel på det grafiske resultatet av denne jobben.

Polygonfilen som er skapt av denne jobben vil bli lagret, og kan følgelig hentes frem som kartografisk grunnlag for senere jobber. Den vil også da bli allokert til enhet 11.

## 5.2. Oppretting av en systemfil med beskrivende data

Dette eksemplet henter inn beskrivende data for de 29 områdene på Prince Edward Island. Dataene konverteres til GIMMS-format og legges ut på systemfil allokert til enhet 13. Filen hvor dataene ligger i "menneskelesbar form" allokeres til enhet 12. Datafilen på enhet 13 katalogiseres og bevares for senere bruk.

På "rådatafilen" på enhet 12 ligger variablene i fritt format. Det vil si at variablene er adskilt av et komma og/eller en eller flere blanke. Data for en enkelt enhet kan også være fordelt på flere linjer. I eksemplet vil det være gitt 30 variabler pr. enhet, men bare 9 av disse er interessante og blir gitt navn. De to første enhetene er gitt på følgende måte - med to linjer for hver:

```
01 3191 1812 6915 4262 3216 2292 -1 10 34 58 60 56 54 11 -1 81 75
    33 32 006 -02 52 12 8 32 12 14 33 43
02 5242 2574 7024 4431 2742 1934 05 5 29 69 69 22 25 6 1 56 56
    15 17 8 23 11 8 13 19 12 18 33 22
```

Rådatafilen vil inneholde data for alle 29 enhetene - altså 58 linjer med data. De 9 første variablene kan beskrives som følger (et kort navn er også lagt på):

- 1 SEQUENCE Enhetens sekvensnummer
- 2 MEMPINC Medianinntekt menn 1978
- 3 FEMPINC Median inntekt kvinner 1978
- 4 MTINCUI Median totalinntekt for "UI beneficiaries" menn 1978
- 5 FTINCUI Median totalinntekt for "UI beneficiaries" kvinner 1978
- 6 MAVUI Gj.snittlig arbeidsløshetstrygd menn 1978
- 7 FAVUI Gj.snittlig arbeidsløshetstrygd kvinner 1978
- 8 MPERTAX % endring i "tax filers" med arbeidsinntekt menn 1978
- 9 FPERTAX % endring i "tax filers" med arbeidsinntekt kvinner 1978

GIMMS-programmet følger:

```
*SYSPARM BATCH
*CREATE DATAFILE FILEIN=11 FILENAME=PEIDATA VARS=50
  TITLE='PRINCE EDWARD ISLAND DATA FILE'
*FILEIN MERGE DATA FILEIN=12 VARS=30
  ABS
  NAMES SEQUENCE MEMPINC FEMPINC MTINCUI FTINCUI MAVUI FAVUI MPERTAX FPERTAX
  BEGIN
*SAVE DATA TO 13
*FILEINFO DATAFILE ALL
*STOP
```

CREATE-kommandoen sørger for at det opprettes en lagerplass for en datafil. Denne bindes ikke til noen fysisk fil, men settes av i maskinens minne. DATAFILE angir at dette skal være en systemfil for beskrivende data og FILEIN=11 henter inn polygonfilen fra enhet 11 for en beskrivelse av hvilke enheter som skal være med. Dataene som senere skal leses inn vil senere bli tilordnet områdene i den rekkefølgen de ligger på polygonfilen. FILENAME og TITLE gir referansenavn på filen på samme måte som tidligere og VARS=50 setter av plass til innlesning av 50 variabler. I og med at polygonfilen vil angi at det er 29 enheter og vi her angir at det skal settes av plass til 50 variabler kan GIMMS sette av tilstrekkelig plass til datamatriksen.

FILEIN-kommandoen sørger for selve innlesningen av rådataene. DATA angir at det er beskrivende data som leses inn og MERGE at datafilen (systemfilen) allerede er opprettet ved hjelp av CREATE-kommandoen og at data skal tilpasses denne strukturen (MERGES med denne). FILEIN=12 angir at rådataene skal leses

fra en fil som er allokeret til enhet 12, mens VARS=30 opplyser GIMMS om at vedkommende rådatafil inneholder 30 variabler. Mens vi under CREATE satte av plass for 50 variabler, leser vi ikke inn mer enn 30. Vi har altså plass til ytterligere 20 variabler som kan fylles inn senere. ABS viser at dataene skal leses som absolutte tall. NAMES gir mulighet til å legge inn navn på variablene. I eksemplet er det lagt inn navn på de første 9 variablene. BEGIN gir programmet beskjed om å begynne å lese data fra input-filen.

SAVE-kommandoen gir ordre om å bevare datafilen på en fysisk fil. Her på filen som er allokeret til enhet 13. Dette blir en GIMMS-systemfil med beskrivende data i GIMMS-format og den vil inneholde de karakteristika som er gitt datafilen gjennom CREATE og FILEIN-kommandoene. Om den fysiske filen som allokeres til FT13F001 katalogiseres og bevares vil den kunne hentes frem igjen som datafil for fremtidige tegninger. Dette forutsettes gjort for de videre eksemplene.

### 6.3. Enkelt koroplettkart med to histogrammer

Til denne jobben forutsettes det at den gamle polygonfilen fra eksempel 1 er bevart og allokeres til enhet 11, og at datafilen som ble skapt i eksempel 2 bevares og allokeres til enhet 13. Videre skal det tegnes grafikk, og en plottefil må derfor allokeres til enhet 19.

```
*SYSPARM DIAGS=1 NOLIST BATCH NOLOGO
*PLOT Parm DEVICE=3
*TEXT Parm ALPHABET 16
*RESTORE DATA FROM 13
*PLOT Prog
*NEWMAP 16 16 FRAME
*GIMMSFILE 11 GENERAL 8
*ORIGIN SHIFT 0 6
*INTERVALS MAVUI USER=1800,1900,2100,2300,2500,3250,FREQ
*MAP MAVUI AREA
*TEXT KEY=TOP,'Dollars' ALPHABET 26
*BARGRAPH SAMEPAGE VAR MAVUI FREQ SHADE CLASSES ORIGIN 8 1
  LENGTH 4 3 NOXLAB NOXTICS NOYAXIS BARTEXT TOP
*TEXT 10 0.5 0.35 'Class histogram' CENTRE CENTVAL=-1
*BARGRAPH SAMEPAGE VAR MAVUI FREQ 20 SHADE ORIGIN 20 1
  LENGTH 8 3 NOXAXIS NOYAXIS BARTEXT TOP
*TEXT 24 0.5 0.35 'Frequency histogram' CENTRE CENTVAL=-1
```

```
*TEXT 14 19 0.6 'PRINCE EDWARD ISLAND  
AVERAGE UNEMPLOYMENT BENEFITS, MALES (1978)'  
SHADE ALPHABET 61 CENTRE CENTVAL=-1  
*END  
*STOP
```

SYSPARM-kommandoen inneholder i tillegg til de sub-kommandoene som er behandlet i eksempel 1 NOLIST og NOLOGO som begge henspeiler på uttegningsfunksjonen. NOLIST fører til at en rekke opplysninger om uttegningsarbeidet ikke blir skrevet ut på jobbens logg-fil, og NOLOGO at GIMMS' logo (egen tekst) ikke blir skrevet ut i kartets nedre høyre hjørne.

Under PLOTPARM-kommandoen velges tegnemaskin 3, som er en Tektronix 4027 grafisk skjerm.

TEXTPARM-kommandoen benyttes for å sette systemparametre omkring tekstutforming. Her er den brukt for å velge alfabet nummer 16. Alfabet betyr i denne forbindelse skrifttype. Skrifttype nummer 16 vil nå bli brukt for tekstutforming såfremt ikke noe annet alfabet angis eksplisitt i TEXT-kommandoene.

RESTORE-kommandoen forstås best om den sees i sammenheng med behandlingen av datafilen i eksempel 2. Der ble det først laget en "intern" datafil med kommandoen CREATE og data ble lagt inn i denne med FILEIN-kommandoen. Deretter ble den interne datafilen lagret på en fysisk fil ved hjelp av SAVE-kommandoen. I jobben i eksempel 3 forutsettes det at den fysiske datafilen nok en gang er allokert til enhet 13. Dataene herfra føres så tilbake til GIMMS' interne datafil med kommandoen RESTORE. RESTORE DATA FROM 13 fører altså til at dataene fra den fysiske filen som er allokert til enhet 13 kopieres inn i den interne datafilen.

PLOTPROG-kommandoen starter opp en avgrenset seksjon av GIMMS-jobben som skal utføre plotting. PLOTPROG avsluttes med en END-kommando, som vi i eksemplet finner 16 linjer lenger ned.

NEWMAP initierer systemet for tegning av et nytt kart. Denne kommandoen må ikke forveksles med kommandoen MAP som faktisk utfører en karttegning. Forskjellen er altså at mens MAP er aktiviserende, er NEWMAP initierende. Under denne kommandoen står 16 16 for billedutsnittets størrelse i centimeter i hhv. X- og Y-retning. Underkommandoen FRAME gir beskjed om at det skal tegnes en ramme omkring dette området.

Kommandoen GIMMSFILE benyttes for å aktivisere en systemfil med lokaliseringdata. I dette tilfellet skal vi benytte polygonbeskrivelsene som er allokert til enhet 11, og det gis videre beskjed om at disse koordinatene skal tegnes

med generaliseringsnivå 8 (forklart i eksempel 1).

Kommandoen ORIGIN benyttes for å justere plasseringen av plottet i forhold til rammen som er satt av på papiret. Tilleggskommandoen SHIFT betyr at koordinatverdiene blir gitt i centimeter, og at kartutsnittets nedre venstre hjørne skal plasseres i det angitte punkt i papirrammen. Her beholdes posisjonen i X-retning, men kartet skyves 6 centimeter oppover arket (skjermen).

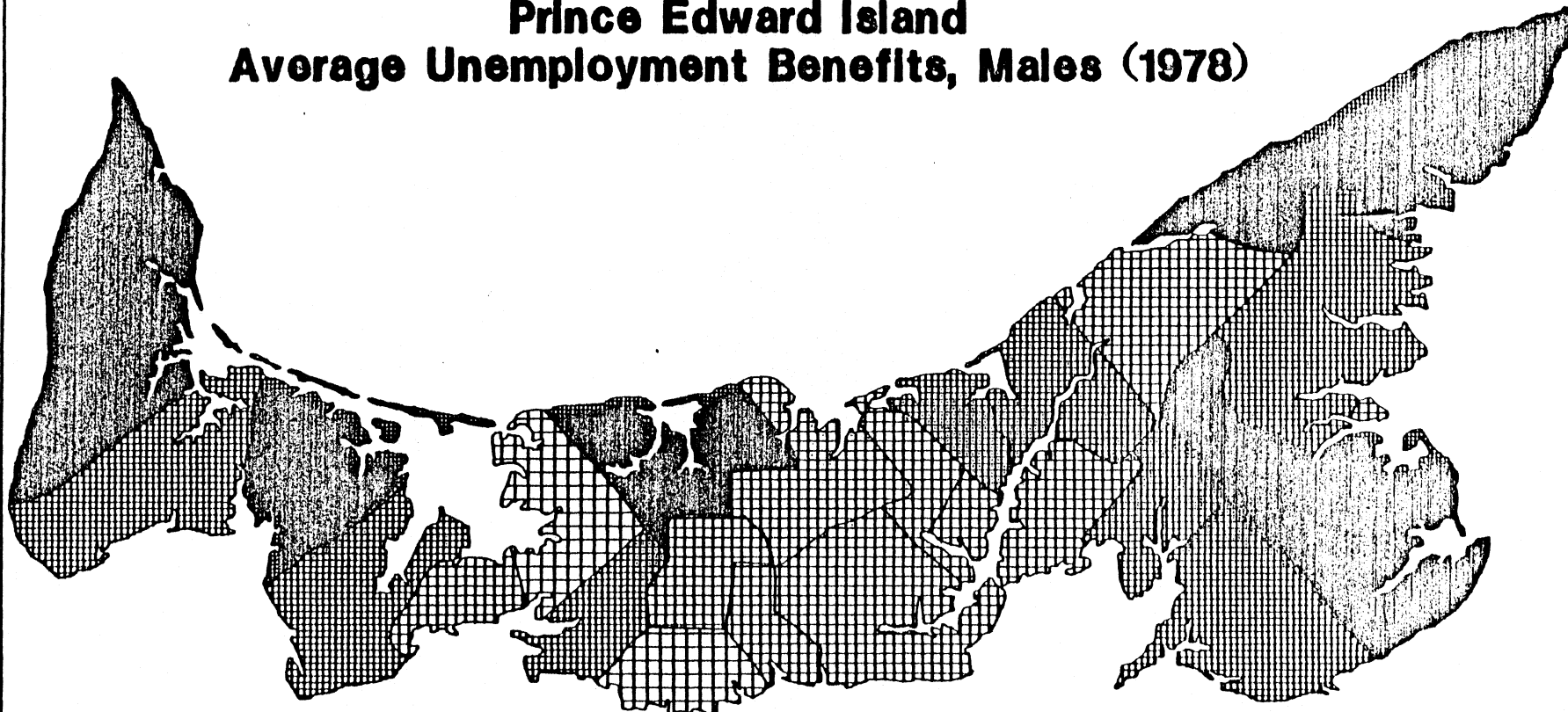
INTERVALS benyttes for å gruppere verdiene på beskrivende variabler med sikte på for eksempel koroplettkart. I eksemplet er det variabelen MAVUI (se eksempel 2) som blir inndelt i klasser. Man kunne også ha skrevet eksplisitt VARIABLE=MAVUI, og ettersom denne variabelene er den 6. av de definerte variablene (se eksempel 2) ville det også være mulig å skrive VARIABLE=6. Spesielt interesserte kan nå legge merke til at det i eksempel 2 ble lest inn 30 variabler på datafilen, men bare satt navn på 9 av disse. Variabler som vi ikke har satt navn på kan vi likevel benytte i utegningsøyemed ved å referere til nummeret de har i den rekkefølgen de blir lest inn. Den aller siste variabelen kan for eksempel kalles frem som VARIABLE=30. Selve klassifiseringen av variabelen kan skje ved ulike statistiske funksjoner, eller ved at brukeren selv setter klassegrensene eksplisitt. I eksemplet er den siste fremgangsmåten valgt, og dette uttrykkes ved USER=. Deretter settes klassegrensene. USER=1800,1900,2100,2300,2500,3250 fører til at variabelen blir delt inn i 5 grupper. En gruppe er områder med verdier fra 1800 til 1900, neste gruppe områder med verdier fra 1900 til 2100 osv. Femte og siste gruppe er områder med verdier fra 2500 til 3250. Subkommandoen FREQ som er hengt på til sist i denne kommandoen gir programmet ordre om å produsere et frekvenshistogram som viser antallet enheter som faller inn i hver av verdiklassene, og skrive ut dette histogrammet på jobbens logg.

Kommandoen MAP aktiviserer kartografifunksjonene i GIMMS og fører til at det faktisk blir tegnet et kart. MAVUI er variabelen som skal benyttes i karttegningen og karttypen er AREA. Vi lar forøvrig i dette eksemplet programmet selv velge ut passende skravurtyper for oss. Merk at koroplettkart i GIMMS-terminologi altså heter AREA.

Kommandoen MAP vil også sørge for at det automatisk blir lagt inn en tegnforklaring på kartet. Denne legges i nedre venstre hjørne. Plassering og utforming av tegnforklaringen er modifiserbar, selv om dette ikke benyttes i eksemplet. GIMMS vil videre huske hvor den har plassert tegnforklaringen - noe som er essensielt å forstå neste kommando:

TEXT-kommandoen benyttes generelt for å legge inn tekst på plottet. I den

# Prince Edward Island Average Unemployment Benefits, Males (1978)



*Dollars*

	2500 -	3250
	2300 <	2500
	2100 <	2300
	1900 <	2100
	1800 <	1900

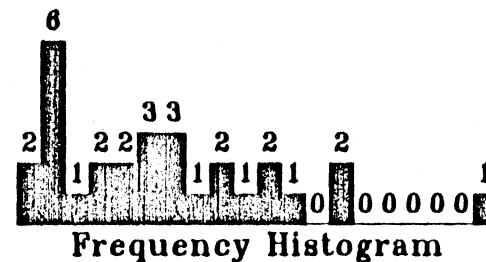
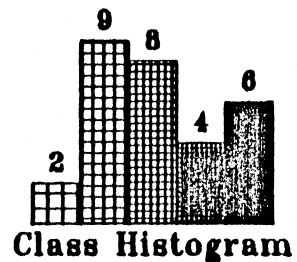


Figure E.5

Figur 10: Koroplett kart tegnet ved hjelp av GIMMS-kommandoene fra eksempel 3 (kap. 6.3).



TEXT-kommandoen som følger direkte etter MAP-kommandoen er posisjoneringen angitt ved KEY=TOP. Dette betyr at teksten skal plasseres relativt til tegnforklaringen (KEY), nærmere bestemt på toppen av denne. Selve teksten er skrevet i gakketegnet ('Dollars') og vi ber dessuten om at alfabet nummer 26 skal benyttes i dette tilfellet. Når vi setter alfabet (skrifttype) eksplisitt under en tekstkommando blir denne skrifttypen benyttet her. Vi endrer med dette imidlertid ikke på systemparameteren som er satt under TEXTPARM tidligere i programmet - den vil fortsatt gjelde. Når vi senere kommer frem til TEXT-kommandoer uten alfabetangivelse er det altså alfabet nummer 16 som vil bli benyttet der.

BARGRAPH er en kommando om å få tegnet ut et histogram. Subkommandoen SAMEPAGE er en kommando om å tegne histogrammet på samme papir som forrige tegning (i dette tilfellet kartet). Histogrammet skal altså tegnes uten sideskift. Når vi i eksemplet tegner på skjerm, vil SAMEPAGE bevirke at skjermen ikke slettes før histogrammet tegnes ut. VAR MAVUI kunne også vært skrevet VARIABLE=MAVUI eller VARIABLE=6 (se under INTERVALS-kommandoen ovenfor). Dette er et utslag av at ortografi og syntaks i GIMMS er nogenlunde romslig. Subkommandoen angir selvsagt hvilken variabel som skal fremstilles i histogrammet. FREQ gir ordre om at dette skal være et frekvenshistogram og SHADE at det skal skraveres. Internt i GIMMS vil vi imidlertid nå ha to forskjellige representasjoner av denne variabelen. Både den originale og den grupperte som vi dannet ved hjelp av INTERVALS-kommandoen. Ved karttegning ble den grupperte versjonen valgt automatisk. Ved histogramuttegning kan vi imidlertid velge. Her velges den grupperte versjonen ved hjelp av subkommandoen CLASSES. Videre angis det hvor histogrammet skal plasseres på skjermen. ORIGIN 8 1 angir at histogrammets nedre venstre hjørne skal plasseres 8 cm. fra venstre kant og 1 cm. opp på skjermen. Sammen med LENGTH beskriver ORIGIN en ramme som histogrammet skal tegnes innenfor. ORIGIN gir rammens nedre venstre hjørne og LENGTH sier at den skal være 4 cm. lang og 3 cm. høy. På denne måten blir histogrammets størrelse fastsatt. NOXLAB, NOXTICS og NOYAXIS er alle negerende kommandoer som fører til at ymse detaljer ikke tegnes ut på histogrammet. Her gjelder det henholdsvis: Tekst på X-aksen (Label), delemerker på X-aksen (Tics) og hele Y-aksen. BARTEXT TOP fører til at GIMMS automatisk genererer histogramtekst og plasserer den på toppen av stolpene. Når vi tegner frekvenshistogram er det antallet enheter i hver gruppe (frekvensen) som blir skrevet ut. Legg merke til at når vi tegner en gruppert variabel på denne måten vil skravuren automatisk bli fordelt som på kartet forøvrig.

Neste TEXT-kommando plasseres eksplisitt av brukeren 10 cm. fra venstre kant og 0.5 cm. oppe på plottet. Dette blir altså et punkt midt under histogrammet. Det tredje tallet (0.35) gir bokstavhøyden i cm. Kommandoen kunne ha vært

utformet POSITION=10,0.5 SIZE=0.35, og på den måten kanskje blitt tydeligere for leseren. CENTRE viser at teksten skal sentreres og CENTVAL=-1 angir at dette skal skje omkring den angitte X-verdien på tekst-linjen (her 10 cm.). Teksten blir altså i dette tilfellet sentrert omkring punktet midt under histogrammet.

Den andre BARGRAPH-kommandoen ligner mye på den første. Vi benytter fremdeles SAMEPAGE for å få histogrammet tegnet på det samme papiret (evt. den samme skjermen). Subkommandoen CLASSES er imidlertid utelatt og vi vil derfor denne gangen generere et frekvenshistogram på grunnlag av de opprinnelige verdiene på MAVUI-variabelen. Histogrammet skal denne gangen tegnes i et rektangel som er 8 cm. langt og 3 cm. høyt (LENGTH 8 3) med nedre venstre hjørne i punktet 20,1. Det skal hverken tegnes X- eller Y-akse, men frekvensene skal skrives ut på toppen av stolpene. Den påfølgende TEXT-kommandoen senterer tekst til histogrammet under det, på samme måten som ved forrige histogram.

Den siste kommandoen av type TEXT benyttes til å legge en overskrift på kartet. Utgangspunktet her er et punkt 14 cm. fra venstre kant og 19 cm. opp på papiret. (OBS: dette er vanvittig - idet vi har laget et totalt tegneutsnitt på 16x16 cm. Misforholdet skyldes at koordinatene til teksten er laget for bruk med plotter, mens tegnerammen er satt for bruk på skjerm uten at tekstkoordinatene er rettet opp...). SHADE betyr at bokstavene skal skraveres. Legg merke til at teksten er skrevet ut på to linjer. På tegningen vil den på samme måte komme ut som to linjer, og begge linjene blir sentrert om det angitte punktets X-verdi. I Y-retning vil imidlertid linje nummer to bli flyttet et passende stykke nedover på arket.

Benytt plottet i figur 10 (s. 31) ved lesning av dette eksemplet. Det viser resultatet av jobben som er gjennomgått i dette underkapittelet.

#### 6.4. Sektordelte flateproposjonale sirkler + 2 histogrammer

Eksempel 4 tegner kart med flateproposjonale sirkler. Disse er videre delt inn i sektorer (kakediagram eller pie-charts). I dette eksemplet bestemmer også brukeren selv hvordan skravuren i sirkelsektorene skal utformes, og GIMMS' databehandlingsfunksjoner tas i bruk for å konstruere nye variable: Sirklene skal vise summen av variablene MEMPINC og FEMPINC, mens sektorene visualiserer hvor stor andel av totalen hver av de to variablene bidrar med.

```
*SYSPARM BATCH NOLIST
*PLOT Parm DEVICE=4
*TEXT Parm ALPHABET 16
*PLOT Prog
  *NEWMAP 30 20 FRAME
  *TEXT POSITION=14,19 SIZE=0.6 TEXT='Prince Edward Island
    Male and Female Employment Income (1978) '
    ALPHABET 61 CENTRE CENTVAL=-1
  *GIMMSFILE 11 GENERAL 6
  *ORIGIN SHIFT 0 6
  *SYMBOLISM AREA
    SEP=0.02 ANGLE=0 PEN=1 & SEP=0.02 ANGLE=90 PEN=1 /
    SEP=0.1 PEN=2
  *MANIPULATE
    *COMPUTE TOTAL ABS=(MEMPINC+FEMPINC)
    *END
  *SYMBOLISM POINT SCALED PIE SIZE 2.5 SHADE
  *SYMBOLISM PIE SHADING 1 2
  *MAP TOTAL MEMPINC FEMPINC POINT
  *BARGRAPH SAMEPAGE VAR MEMPINC,FEMPINC GROUP 2 SCALED SHADEGRP 1 2
    NOXLAB NOXTICS NOXSTICS ORIGIN 1.5 1 LENGTH 10,5 YUNITS=20 YMAX=100
    YMIN=0 YTITLE='Percent' YTITLEHT=0.3 XTITLEHT=0.3
    XTITLE='Percentages by communities'
  *BARGRAPH SAMEPAGE VAR MEMPINC,FEMPINC GROUP 2 ADDITIVE SH_DEGRP 1 2
    NOXLAB NOXTICS NOXSTICS ORIGIN 14 1 LENGTH 10,5 YTITLEHT=0.3
    YTITLE='Dollars' XTITLEHT=0.3 XTITLE='Values by communities'
  *TEXT KEY=1 SIZE=0.3 TEXT='Male Employment
    Income'
  *TEXT KEY=2 SIZE=0.3 TEXT='Female Employment
    Income'
  *END
*STOP
```

En rekke av kommandoene i dette programmet er kjente fra tidligere eksempler. Disse vil i hovedsak ikke bli kommentert her. Programmet åpner med å sette ulike system-, plote- og tekst-parametre samt hente inn datafilen. I plote-programmet defineres tegneområdet og det skrives på overskrift allerede før koordinatfilen defineres. Den første nye kommandoen er:

SYMBOLISM-kommandoen som i dette programmet finnes i tre utgaver. Den første av disse har subkommandoen AREA og definerer symboler til bruk i skravering

av flater. I eksemplet dreier det seg om flatene i sirkelsektorene, men slik flateskravurdefinisjon kan også benyttes for å lage egne skravurer til bruk på koroplettkart. I eksemplet defineres det to skravurer, og disse skilles ad med tegnet "/". Det er mulig å definere enda flere skravurer, men alle skravurene må skilles ad med "/". Skravurene blir automatisk nummerert slik at den første av de to skravurene i eksemplet blir nummer 1, den andre nummer to.

Legg videre merke til at den første av de to skravurene er definert med to ledd, adskilt av tegnet "&". I hvert av disse leddene blir det beskrevet et eget mønster, og tegnet "&" betyr at disse to mønstrene skal legges oppå hverandre i den endelige skravuren. Den andre skravurdefinisjonen består bare av ett enkelt mønster, og definisjonen har følgelig kun ett ledd.

Hver ledd - eller hver mønsterdefinisjon - i en skravurdefinisjon består av inntil 3 parametre. Kun det første av disse (SEP) er nødvendig. SEP står for separasjon og forteller hvor stor avstand det skal være mellom skravur-linjene i mønsteret. Avstanden angis i cm. ANGLE angir vinkelen skravur-linjen skal tegnes med i forhold til horisontalen. Vinkelen gis i grader. Om det ikke angis noen vinkel tilsvarende det ANGLE=0. I skravur nr. 2 vil derfor linjene bli tegnet horisontalt. I skravur nr. 1 blir linjene i det ene mønsteret tegnet horisontalt, og linjene i det andre mønsteret tegnet 90 grader på horisontalen: Altså vertikalt. Når disse to mønstrene legges ovenpå hverandre oppstår et rutemønster. Her tegne imidlertid linjene så tett (avstand 0.02 cm.) at mønstrene tilsammen blir helt svart..Den tredje parametren i en mønsterbeskrivelse er PEN. Denne er heller ikke nødvendig. På plottere sier den hvilken penn som skal benyttes når skravuren tegnes. For eksempel har Tektronix 4663 to penner, og ved bruk av denne kan vi velge mellom PEN=1 og PEN=2. På grafiske skjermer vil PEN være en måte å velge farge på. Har en skjerm 8 farger, vil vi kunne velge farge nummer 8 med PEN=8. Det opp til den som programmerer koblingen mellom GIMMS og en tegne-maskin å definere hvilke farger som er hvilke nummer. Om man unnlater å spesifisere penn, eller velger en som ikke finnes skal den pennen (eller fargen) som er penn nr. 1 bli brukt.

Neste SYMBOLISM-kommando har subkommandoen POINT og definerer punktsymboler. Punktsymboler er symboler festet til et punkt på kartet: Sirkler, kakediagram (sektordelte sirkler), firkanter, stjerner etc. I eksemplet velges kakediagram (PIE) og disse skal skaleres (størrelsen skal varieres flateproposjonalt). At det skal skaleres angis med SCALE. Når MAP-kommandoen senere starter selve tegningen vil skaleringen av symbolene bli utført ved hjelp av den første av de variablene vi anfører på denne kommandoen. SHADE sier at

symbolene skal skraveres. Siden symbolene er av type PIE må imidlertid også et tredje kort med SYMBOLISM-kommando til for å få til dette, noe vi kommer tilbake til. Den siste subkommando på innværende linje er SIZE 2.5 som definerer størrelsen på det største av kakesymbolene. Størrelsen gis i cm. og jeg tror det er snakk om arealet på største sirkel - altså  $2.5 \text{ cm}^2$  - uten at dette er spesifisert i manualen. De øvrige sirkelflatene vil bli skalert i forhold til denne største sirkelen.

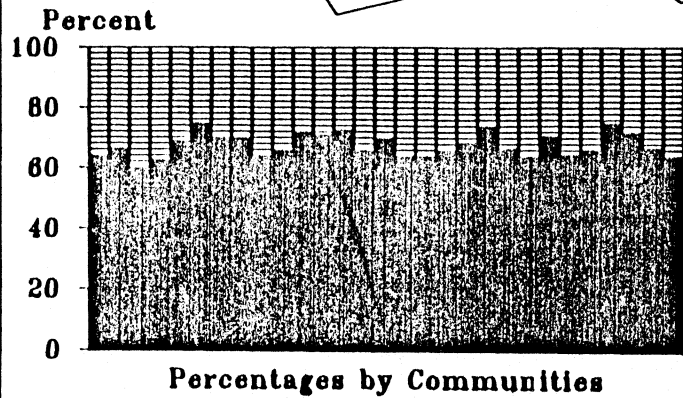
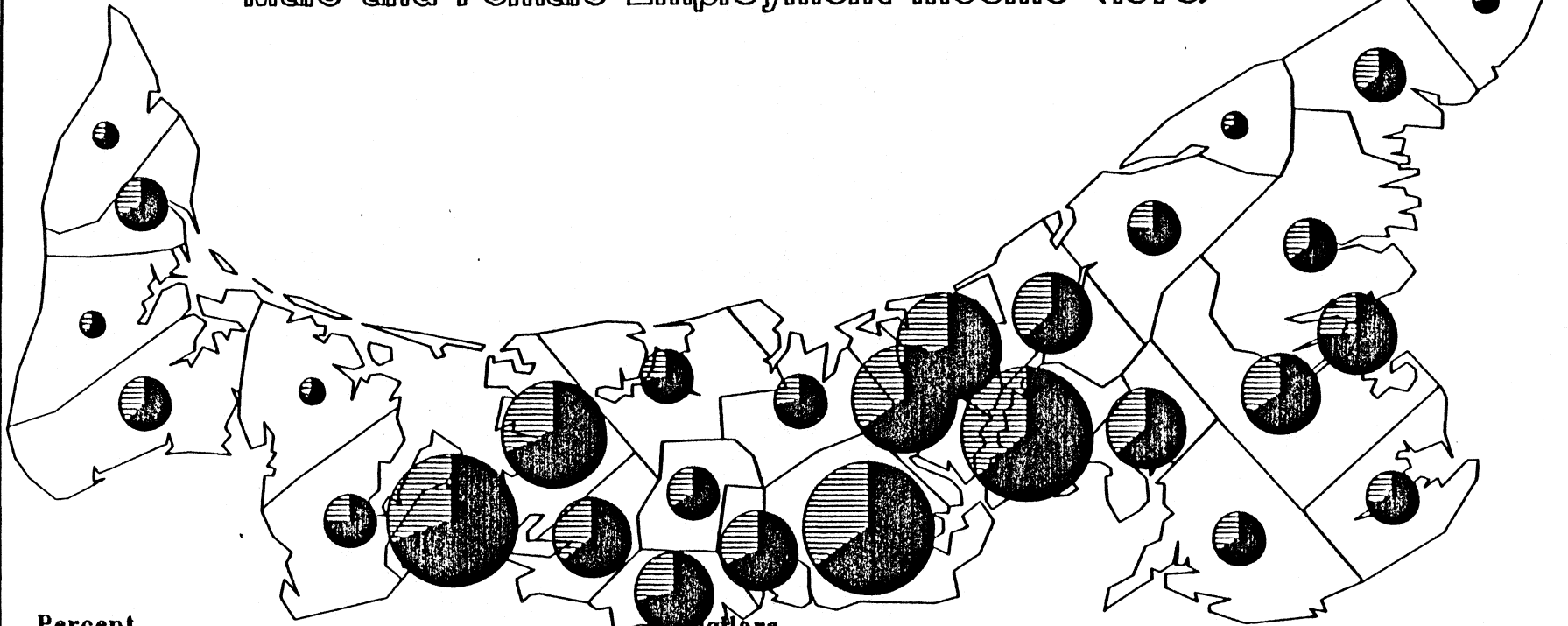
Den tredje SYMBOLISM-kommando har subkommando PIE og benyttes for å definere nærmere hvordan skraveringen av sektorene i kakene skal utføres. SHADING 1 2 viser tilbake til kommando SYMBOLISM AREA der skravur nummer 1 og 2 er definert, og tilordner disse til variablene på det etterfølgende MAP-kortet. Vi kan nå se de tre SYMBOLISM-kommandoene og MAP-kommando i sammenheng:

- MAP-kommandoen aktiviserer tegnedelen av GIMMS. POINT angir at det skal tegnes kart med punktsymboler, og at variablene som skal benyttes er TOTAL, MEMPINC og FEMPINC.
- Under SYMBOLISM POINT er det nærmere beskrevet hva slags punktsymbol som ønskes: Det skal være kakediagram, og kakene skal skaleres. Den første av variablene på MAP-kortet blir automatisk brukt til skaleringen. De øvrige deler kakene inn i sektorer.
- Under SYMBOLISM PIE sies det at sektorene skal skraveres. Sektoren for den første av sektorvariablene med skravur nummer 1, sektoren for den andre av sektorvariablene med skravur nummer 2.
- De to skravurtypene vi henviser til på SYMBOLISM PIE kortet er definert på kommando SYMBOLISM AREA

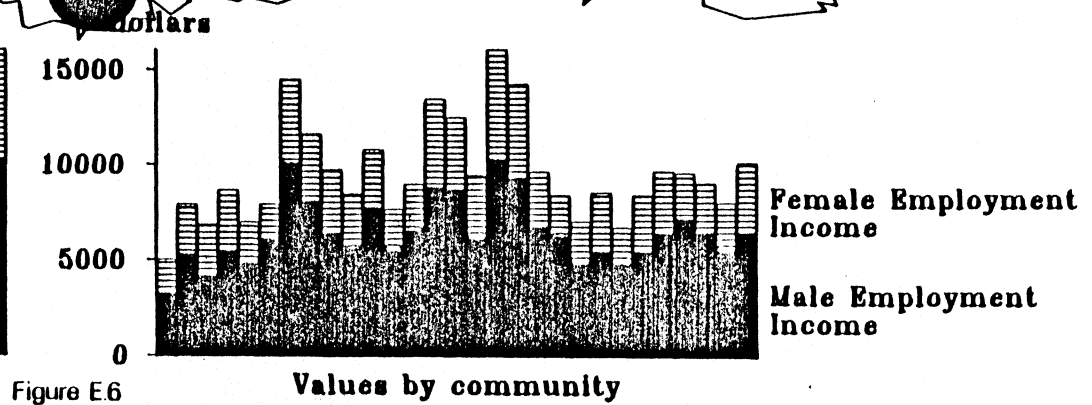
Våkne lesere vil nå ha merket seg to forhold. Vi har lettvent hoppet over kommando MANIPULATE, samtidig som MAP-kommando angir at symbolene skal skaleres ved hjelp av en variabel ved navn TOTAL som ikke er definert i eksempel 2. Denne variabelen konstrueres i MANIPULATE-kommando. I eksemplet ser man at MANIPULATE utgjør en egen programblokk som innledes med kommando MANIPULATE og avsluttes med kommando END. Inne i denne blokken benyttes en kommando COMPUTE for å skape en ny variabel som gis navnet TOTAL. Den settes sammen av summen av de to variablene MEMPINC og FEMPINC som finnes fra før i datasettet. ABS sier at dette skal være et absolutt tall.

Under kommando BARGRAPH finner vi nå at det er kommet til en subkommando NOXTICS som betyr at det ikke skal settes på sub-tics, som er en enda finere nedstipling av aksene enn tics. Denne er antagelig ikke nødvendig når vi benytter NOXTICS.

# Prince Edward Island Male and Female Employment Income (1978)



Percentages by Communities



Values by community

Figure E.6

Figur 11: Kart med kakediagram tegnet ved hjelp av GIMMS-kommandoene fra eksempel 4 (kap. 6.4).

Figur 11 (s. 37) viser også hvordan histogrammene i dette plottet er satt sammen av flere variabler. Dette oppnår vi ved å oppgi to variabler på BARGRAPH-kommandoene: VAR MEMPINC,FEMPINC. Når det i tillegg angis SHADEGRP 1 2 vil den første av variablene bli skravert med skravurtype 1 og den andre med skravurtype nummer 2. Nummereringen av skravurtypene viser også her tilbake til kommandoen SYMBOLISM AREA. Det er også nødvendig å angi at det er to variabler ved hjelp av GROUP 2. De to forskjellige utformingene av histogrammene får vi ved subkommandoene SCALED (venstre graf) og ADDITIVE (høyre graf). SCALED fører til at alle søyler for en enkelt variabel skaleres relativt til den lengste søylen for vedkommende variabel. ADDITIVE er en ordre om at de to søylene skal legges til hverandre uten transformasjon.

#### 6.5. Kurvediagram med 6 kurver

GIMMS er ikke et rent kartografiprogram, men kan også benyttes til å tegne annen grafikk - så som kurver, histogrammer og kaker. Dette er i noen grad vist i de to foregående eksemplene der det er tegnet inn databeskrivende histogrammer på kartene. Eksempel 5 vil derimot utelukkende være et kurvediagram med fremstilling av 6 variabler i kompatibel form.

```
*SYSPARM BATCH NOLOGO
```

```
*PLOTARM DEVICE=4 FACT=0.8
```

```
*GRAPHICS
```

```
*TEXTARM ALPHABET 15
```

```
*GRAPH VARS=MEMPINC,FEMPINC,MTINCUI,FTINCUI,MAVUI,FAVUI GROUP=6
```

```
PAGESIZE=30,20 ORIGIN=2,2 LENGTH=20,15
```

```
NOXLAB XUNITS=1 NOXTICS XTITLEHT=0.4 XTITLE='Values by Community'
```

```
YMIN=1000 YMAX=10100 YTITLEHT=0.4 YTITLE='Dollars'
```

```
LINEGRP=0,9,0,9,0,9
```

```
THICKGRP=2,2,5,5,2,2
```

```
SYMBGRP=0,0,0,0,2,2
```

```
*TEXT POSITION=15,19 SIZE=0.6 TEXT='Prince Edward Island
```

```
Employment and Unemployment Income' ALPHABET=61 CENTRE ONX
```

```
*TEXT SIZE=0.3 KEY=1 '©Employment Income (Males)'
```

```
/SIZE=0.3 KEY=2 '©Employment Income (Females)'
```

```
/SIZE=0.3 KEY=3 '©STotal Income
```

```
(UI beneficiaries, Males)'
```

```
/SIZE=0.3 KEY=4 '©STotal Income
```

```
(UI beneficiaries, Females)'
```

```
/SIZE=0.3 KEY=5 '@SUI benefits (Males)'  
/SIZE=0.3 KEY=6 '@TUI benefits (Females)'
```

\*END

\*STOP

GRAPHICS-kommandoen er en initierende kommando som starter opp en egen modul rettet mot grafikk-generering. Modulen avsluttes med kommandoen END som vi finner igjen som den nest siste kommandoen i programmet. Den aktiviserende kommandoen i modulen er GRAPH hvor selve tegnearbeidet utformes. GRAPH er i seg selv en kommando som gir beskjed om at den typen figur man ønsker er et kurvediagram. Parametrene til subkommandoen VARS sier hvilke variabler som skal være med i diagrammet, mens GROUP angir hvor mange variable det er. PAGESIZE definerer hvor stort papirutsnitt tegningen skal benytte, mens ORIGIN og LENGTH på samme måte som for BARGRAPH-ene i tidligere eksempler setter av en "boks" inne i hvilken selve diagrammet skal tegnes. XUNITS er hvilke enheter som skal benyttes på X-aksen, mens YMIN og YMAX definerer verdi-området som skal dekkes av Y-aksen.

De tre siste sub-kommandoene på denne hovedkommandoen tar alle 6 parametre - altså like mange parametre som vi skal ha med variabler i grafen. Disse tre subkommandoene definerer kurvene variablene fremstilles gjennom.

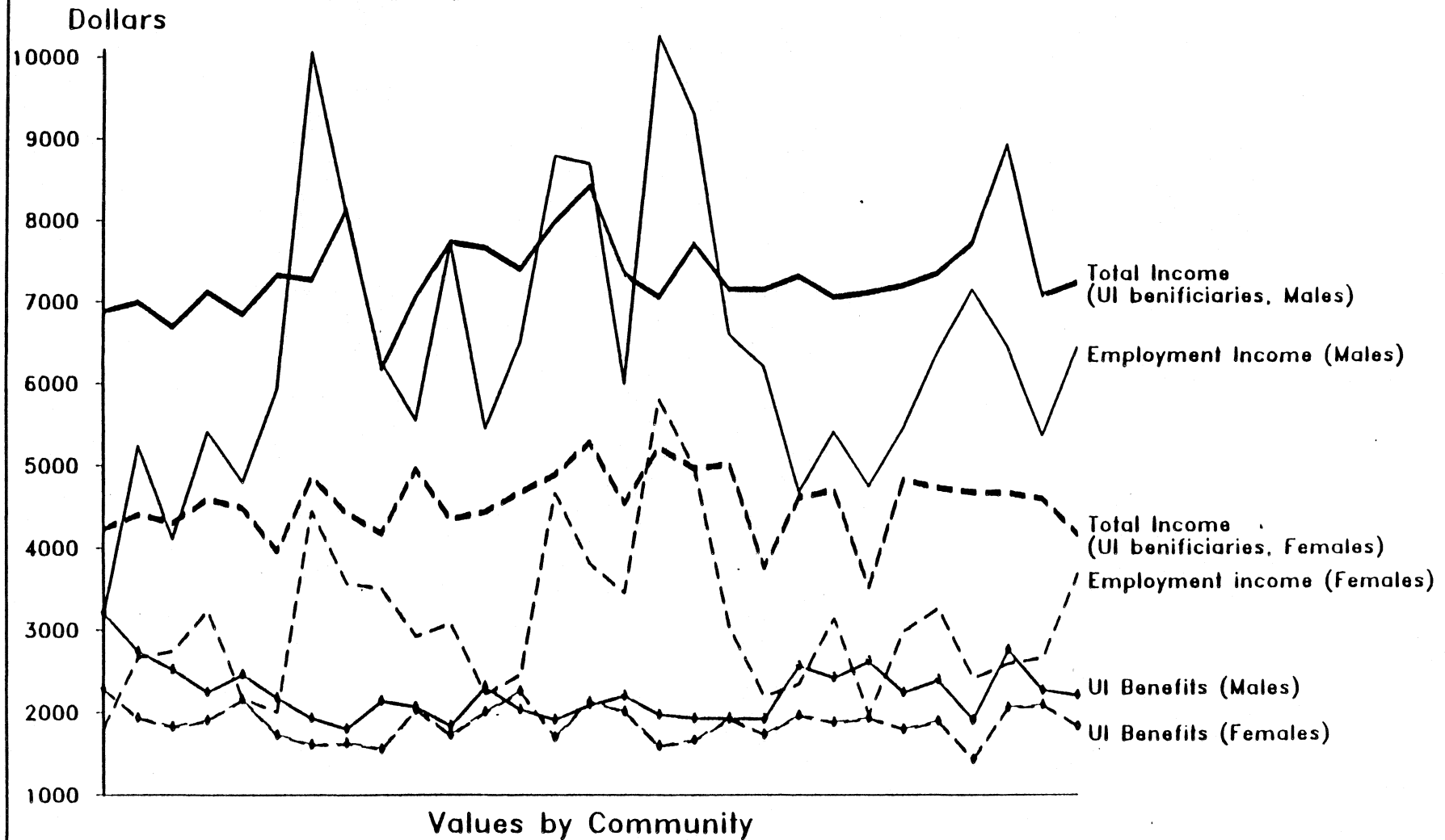
LINEGRP er linjetyperne som skal benyttes for de 6 variablene. De tilgjengelige linjetyperne er beskrevet på side 6.78 i GIMMS-manualen. THICKGRP setter på samme måten tykkelsen på de seks kurvene. Tykke linjer oppnås ved at kurvene blir streket opp flere ganger, og samtidig forskyves en smule. Linjetykkelse 2 oppnås ved å trekke kurven to ganger, mens linjetykkelse 5 er et resultat av at kurven strekes opp 5 ganger. Den siste versjonen blir følgelig noe tykkere. SYMBGRP benyttes for å velge symboler som skal markere datapunktene langs de forskjellige kurvene.

I den første av TEXT-kommandoene benyttes sub-kommandoen ONX. Dette tilsvarer antagelig den tidligere bruken av CENTVAL=-1 og gir altså beskjed om at teksten skal sentreres om den X-verdien som er gitt i POSITION-subkommandoen.

Den andre TEXT-kommandoen benyttes for å henge forklarende tekster i enden av kurvene. De 6 tekstene (en for hver kurve) er adskilt av tegnet "/". Bokstavstørelsen gis med subkommandoen SIZE og plasseringen med subkommandoen KEY. Denne måten å plassere tekst på kan benyttes sammen med kommandoene LEGEND, BARGRAPH og GRAPH. Ved bruk av BARGRAPH og GRAPH vil programmet benytte forskjellige typer symboler - i inneværende eksempel 6 linjesymboler - og disse holder programmet orden i ved å nummerere dem. Vi plasserer tekst ved siden av det stedet et symbol er brukt i en GRAPH eller BARGRAPH ved å vise til symbol-



# Prince Edward Island Employment and Unemployment Income



Figur 12: Kurvediagram tegnet ved hjælp av GIMMS-kommandoene i eksempel 5, (kap. 6.5).

nummeret gjennom KEY=nummer.

Inne i tekst-strengene (i apostrof) som skal skrives ut er det i eksemplet benyttet et nabla-tegn -@. Dette blir ikke skrevet ut, men betyr at neste bokstav skal leses som en tekstformatterende kommando. Hverken nabla-tegnet eller påfølgende tegn blir altså skrevet ut (en sannhet med modifikasjoner). I eksemplet starter de to første tekststrengene med@T, deretter følger to strenger som begynner med@S og nok en streng som begynner med@T. Ved bruk av@T blir resten av strengen skrevet en halv bokstavhøyde under linjen hvor den ellers ville stått. I figuren som følger eksemplet ser vi hvordan disse tekstene er senket noe i forhold til kurvene de står utenfor. Ved bruk av@S skrives derimot strengen ut en halv bokstavhøyde over linjen, slik at disse to linjene - den første og den tredje i figuren starter noe i overkant av kurvene de er plassert utenfor. Disse kontrolltegnene og flere med er beskrevet i GIMMS-manualens appendix A.

Figur 12 (s. 40) presenterer resultatet av den GIMMS-jobben som er gjennomgått i dette underkapittelet.

## 7. DIVERSE FORHOLD VED BRUK AV GIMMS

Dette kapittelet samler opp en serie mindre forhold man bør kjenne ved bruk av GIMMS på Byråets IBM-anlegg. Delvis er dette fenomener som skyldes implementasjonen av programmet på IBM, delvis er det forhold knyttet til de ferdige koordinatdatasettene som er lagt inn.

### 7.1 VALG AV TEGNEMASKIN

Valg av tegnemaskin fungerer anderledes på IBM enn hva som er beskrevet i GIMMS-manualen. I den foreliggende versjonen av GIMMS velges tegnemaskin ved hjelp av subkommandoen DEVICE under +PLOT Parm. Denne subkommandoen tar et heltall som parameter. Heltallet skal være nummeret på den tegnemaskintypen man ønsker å generere kode for. Pr. 1. september 1985 er 4 koder tilgjengelige:

- 1 - Dumpdriver
- 2 - Linjeskriver
- 3 - Tektronix 4027 A grafisk skjerm
- 4 - Tektronix 4663 grafisk tegnebord

For å velge å generere grafisk kode for Tektronix-skjermen må man dermed gi følgende kommando i i jobbens kommandofil:

```
*PLOT Parm DEVICE=3
```

En kort beskrivelse av de ulike typene følger:

Dumpdriver (Devicenummer 1) er ingen tegnemaskin, men en kode som fører til at alle tegnekommandoer fra GIMMS skrives ut som tekst. Den kan for eksempel tas ut på jobbens logg-fil (Sysout). Dumpdriveren vil først og fremst være interessant for systemprogrammerere, men kan benyttes for å finne generelle tegnefeil som skyldes GIMMS og ikke de enkelte driverene.

Linjeskrivergrafikk er upresentabel og har meget dårlig oppløsning. Men samtidig er den rask å generere. Når nye kartgrunnlag er lagt inn, eller ved utprøving av omskalering, rotasjon og flytting av figurer, kan denne muligheten benyttes for å få et første overblikk over resultatet. Linjeskrivergrafikk kan enklest tas ut gjennom jobbens logg-fil (sysout).

At resultatene av såvel dumpdriver som linjeskrivergrafikk enklest tas ut via sysout fra jobben innebærer at man i begge tilfeller allokere fil 19 til denne i jobbens JCL:

```
//FT19F001 DD SYSOUT=* 
```

Kartet (eventuelt dump'et) får man så ut ved å sende loggen fra sysout-held-køen til en eller annen linjeskriver.

De to Tektronix-maskinene (devicenummer 3 og 4) står begge på 12.kontor i 3.etg. i Jernbanegata - Kongsvinger. For å tegne på disse må koden legges ut på en egen fil allokert til FT19F001. Dette er beskrevet på side 9 i dette notatet. De to tegnemaskinene er begge koblet opp som Remote printer nummer 13 (RMT13) på IBM. Filene sendes til denne ved hjelp av TSO-kommandoen PRINTO, eller ved hjelp av "Route to local printer"-faciliteten i ISPF's meny nummer 3.6. Dette gjelder altså både filer med kode generert for skjerm og tegnebord - systemet finner selv ut av hvilken av de to maskinene tegningen skal lages på såfremt rett devicenummer er brukt på PLOTPARM-kommandoen i GIMMS.

## 7.2 KOMMANDOFILEN

GIMMS leser alltid 80 kolonner på hver linje i kommandofilen. Samtidig har editoren i ISPF en tendens til å generere linjenummere i kolonne 72-80. Om disse ikke fjernes vil de bli lest som kommandoer av GIMMS, noe som gjerne gir kaotisk resultat. Den automatiske linjenummereringen slås av inne i editor med kommandoen NUMBERS OFF. Iblant må nummerene også fjernes eksplisitt av brukeren.

GIMMS utføres i batch-modus - dvs. at jobbene submittes til systemet. Dette legger endel begrensninger på hvor datasett som jobben skal benytte kan befinne seg. Vi kan skille mellom TSO-datasett på den ene siden, og produksjons- og test-datasett på den andre siden. TSO-datasett er datasett som har en brukerident som første ledd i navnet, f.eks LK22GHS. Test- og produksjonsdatasett har fire tegn i første ledd og begynner på TK eller P. I en batch-jobb er det kun den filen som inneholder jobbens JCL som får lov til å ligge på et TSO-datasett. Om GIMMS-kommandoene legges inn in-stream (se s. 9) på denne filen vil det fungere greit. Om GIMMS-kommandoene derimot skal ligge på en egen fil, må denne være en del av et test-datasett (TK-sett).

Test-datasett kan editeres på vanlig måte fra ISPF, men dannelsen av dem kan kun skje i en batch-jobb. Figur 13 inneholder den nødvendige JCL for å opprette et TK-sett. Legg merke til den fjerde parameteren på SPACE-kommandoen (her 3) som setter av plass til directory og fører til at datasettet blir partisjonert og kan inneholde flere medlemmer. Om den var 0 eller blanket ville man fått et udelt datasett.

```
//LK22GHSN JOB (6155),'GHS',MSGCLASS=X,NOTIFY=LK22GHS,  
//          MSGLEVEL=(0,0),CLASS=A  
//ALLSTEP EXEC PGM=IEFBR14  
//LAGNYTT  DD DSN=TK22.S6155.GHS.GIMMS,DISP=(NEW,CATLG,DELETE),  
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=4080),UNIT=DISK,  
//          SPACE=(4080,(10,10,3)),VOL=SER=SSB002  
//
```

Figur 13: JCL for å allokere et nytt TK-datasett.

### 7.3 FERDIGE GIMMS-DATASETT

Fire polygoniserte koordinatdatasett er lagt inn og klargjort for bruk mot GIMMS. Disse er:

SSB1.GIMMS.KONGKRTS - Kongsvinger inndelt i 80 grunnkretser

SSB1.GIMMS.KONGDOMR - Kongsvinger inndelt i 5 delområder

SSB1.GIMMS.NORGEKOM - Norge inndelt i 454 kommuner

SSB1.GIMMS.NORGEFYL - Norge inndelt i 19 fylker

Alle datasettene er lagt inn med 9 + 1 definerte generaliseringsnivåer (se nærmere forklaring side 25):

GENERAL=0 gir kun linjer mellom segmentenes endepunkter

GENERAL=1 gir et svært forenklet kart

GENERAL=9 gir et svært detaljert kart

Generaliseringsnivåene mellom 1 og 9 gir en gradvis overgang fra det mest enkle til det mest kompliserte kartet. Valg av nivå må gjøres ut ifra hva kartet skal benyttes til, og hvilken tegnemaskin man bruker. Merk imidlertid at de kompliserte/detaljerte kartene også tar lengst tid å tegne.

Siden GIMMS arbeider i centimeter er målestokkene følgende:

- Kongsvingerkartene er gitt i 1:100

- Norgeskartene er gitt i 1:100000

Hjørnekoordinatene for de rektanglene som omslutter kartene er i kartkoordinater: gjengitt i figur 14.

	SYD-VEST		NORD-ØST	
	X	Y	X	Y
Kongsvinger	640000	6640000	710000	6710000
Norge	0	0	12000	15200

Figur 14: Hjørnekoordinater (i kartkoordinater) for de ferdig innlagte polygondatasettene.

Begge datasettene er bygget opp fra koordinater gitt i UTM-33. Kongsvingerdataene er ukonverterte meterkoordinater i dette systemet, mens Norge-koordinatene er kilometerkoordinater påplussset 800 horisontalt og fratrukket 64400 vertikalt. Nye elementer på Kongsvingerkartene kan dermed legges rett inn med verdier fra samme koordinatsystem, mens slikt arbeid for Norge-kartene krever bearbeiding. Koordinatene må da for det første samles inn som kilometerkoordinater, og videre transformeres etter følgende enkle formel:

$$\begin{aligned} X_1 &= X_0 + 800 \\ Y_1 &= Y_0 - 64400 \end{aligned}$$

Der  $X_1$  og  $Y_1$  er de transformerte koordinatene, mens  $X_0$  og  $Y_0$  er de innsamlede kilometerkoordinatene.

#### 7.4 PLASSERING PÅ PAPIRET

GIMMS er istand til automatisk i tilpasse alle kart til papir eller skjerm. Rektangelet omkring det polygoniserte kartet definert ved LIMITS blir da transformert inn på billedflaten. Iblant ønsker man dog å bestemme målestokk og plassering selv.

Målestokken settes med GIMMS-kommando \*SCALE og en subkommando FACTOR, eksempel:

```
*SCALE FACTOR=0.0004
```

Skaleringsfaktoren beregnes ut ifra målestokken kartets koordinater er gitt i ( $O_m$ ) og den ønskede målestokken (M), slik:

$$\text{Faktor} = M/O_m$$

Figur 15 viser fremgangsmåten for å få et Kongsvingerkart i målestokk 1:250000.

Skaleringsfaktor for å tegne Kongsvinger i 1:250000

Faktor =  $M/O_m$

$M = 1:250000$

$O_m = 1:100$

Faktor =  $(1:250000):(1:100) = 0.000004:0.01 = 0.0004$

GIMMS-kommandoen blir:

\*SCALE FACTOR=0.0004

Figur 15: Omregningsarbeid for å finne skaleringsfaktoren som tegner Kongsvingerkart i målestokk 1:250000.

Den mer direkte plasseringen av plottet på papiret skjer gjennom GIMMS-kommandoen \*ORIGIN. Denne har to subkommandoer MAP og PLOT benyttes til å plassere tegningen. Begge subkommandoene tar en X- og en Y-koordinat som parametre:

\*ORIGIN MAP=xkor,ykor PLOT=xkor,ykor

Om en av subkommandoene utelates er det det samme som å angi den med koordinatene 0,0. Under PLOT angir man et punkt på papiret/skjermen i centimeter fra nedre venstre hjørne. Under MAP angir man et punkt på originalkartet. Ved uttegning blir disse to punktene "lagt sammen" og man får en eksplisitt plassering av kartet på papiret.

Siden Kongsvingerkartets nedre venstre hjørne er gitt med koordinatene 640000,6640000 (se fig. 14) passes dette kartet inn på papiret med

\*ORIGIN MAP=640000,6640000 PLOT=0,0

De angitte koordinatene på kartet plasseres da i plottets nedre venstre hjørne (0,0). Om vi vil skyve plottet 3 cm. mot høyre og 2 cm. opp på papiret blir kommandoen:

\*ORIGIN MAP=640000,6640000 PLOT=3,2

Videre lar plottet seg dreie med kommandoen +ROTATE som tar roteringsvinkelen direkte som parameter angitt i grader. Dreiningen skjer mot urviseren. 90 graders dreining angis følgelig med:

\*ROTATE=90