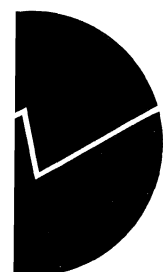


Rune Johansen

**REGARD –
Modell for regional analyse av
arbeidsmarked og demografi**
Teknisk dokumentasjon

Notater



Innholdsfortegnelse

I	Katalogstruktur og Filoversikt	3
1	Katalogstruktur	5
2	Filoversikt	5
2.1	Data og koeffisienter	6
2.2	Macroer	8
2.3	Resultat-filer	11
II	Innhenting av data. Styringsmacroer	13
1	Innhenting av FNR-tall	15
2	Tall fra MODAG/KVARTS	16
2.1	Eventuelle 'manuelle' justeringer	16
3	Styringsmacroen regard	17
3.1	Hjelpemacroen kopi	18
III	Formodellen. Yrkesprosjenter	21
1	Innledning	23
2	Variable i Formodellen	23
3	Macroene i Formodellen	24
3.1	Macroen fylkvis	24
3.2	Macroene invest2, lagseq og syss	26
3.3	Macroen arbstyrk	26
3.4	Aggregering fra fylke til region	26
3.5	Beregning av yrkesprosjenter	26
4	Analyse av resultater. Beregning av vekstprosjenter	27
IV	Modellen ARBEID	29
1	Innledning	31
2	Variable i ARBEID	31
2.1	Endogene variable	32
2.2	Eksogene variable	33
2.3	Koeffisienter	33
3	Generering/endring av modellen	34
4	Likningsstrukturen i ARBEID	34
4.1	Litt om generering av likninger i TROLL	35
4.2	Macroen nysyssutd	35
4.3	Macroen gam2	36
4.4	Macroen ettersp2	36

4.5	Macroen nyutd	37
4.6	Macroen flyttmod	38
4.7	Macroen flytmod2	39
4.8	Macroen flytmod5	40
4.9	Macroen innutv	40
4.10	Macroen nysyss	41
4.11	Macroen nytilbud	41
V	Demografiske oppdateringer	43
1	Innledning	45
1.1	Hjelpemacroene colsplit og kombiner	45
2	Variable i REGARD's demografidel	46
2.1	Inn- og ut-variable	46
2.2	Mellomregninger	48
2.3	Hjelpevariable	50
3	Macroene i demografidelen	52
3.1	Macroen ninv	53
3.2	Macroen flytt	55
3.3	Macroen newborn	56
3.4	Macroen innbef	57
3.5	Macroen rb	58
3.6	Macroen utbef	58
VI	Eksport av resultater	61
1	Innledning	63
2	Macroer	63
2.1	Macroen exbef	63
2.2	Macroen arbst	64
2.3	Macroen sysjr	64
2.4	Macroen xml	65
A	Lister og aggregeringer i REGARD	67
1	Aldersgruppene i REGARD	67
2	Sektorer i MODAG/KVARTS. Aggregering fra FNR	67
2.1	Aggregering fra fylkesvist nasjonalregnskap (FNR)	69
3	Fylkeskoder og regioninndelingen i REGARD	72
B	Ettårig versjon av REGARD	75
1	Innledning	75
2	Macroene	76
2.1	Macroen ninv	76
2.2	Macroen flytt	77
2.3	Macroen newborn	77
2.4	Macroen rb	78

Innledning

Dette notatet inneholder den tekniske dokumentasjonen for den nye versjonen av regionalmodellen REGARD, utviklet ved Forskningsavdelingen i Statistisk sentralbyrå i perioden fra høsten 1995 til sommeren 1997. Arbeidet med omleggingen har vært ledet av forskningssjef ved seksjon for offentlig økonomi og personmodeller, Nils M. Stølen. Forfatteren av dette notatet har gjort mesteparten av programmeringsarbeidet. Ellers har Lasse Stambøl, Turid Åvitsland og Eva Ivås vært sterkt involvert i prosjektet, og gitt viktige bidrag til modellens utforming. Figurene vedlagt dette notatet er laget av Eva Ivås.

Nye REGARD er delvis en oversettelse av den gamle modellen fra 'gammel' TROLL på IBM stormaskin til *portabel* TROLL på UNIX arbeidsstasjon, men den inneholder også noen strukturelle endringer fra det gamle opplegget. Delmodellen med likninger for varekryssløpet (PRODINNT) er nå fjernet. Dessuten er den såkalte formodellen vesentlig forenklet. I det nye systemet består formodellen av et sett med 'rene' beregninger, hvor nasjonale tall for bruttoprodukt, investering og sysselsetting blir splittet opp i regionale verdier. Yrkesprosentene blir også beregnet i formodellen. Også den gjenværende *modellen*, ARBEID, er noe forenklet sammenliknet med det gamle opplegget. Selv om PRODINNT er fjernet, har vi delvis oversatt macroene for generering av likningene i denne modellen fra gamle TROLL. Disse ligger derfor klare dersom man senere skulle ønske å inkludere PRODINNT på nytt i REGARD. Noe tilrettelegging, samt retting av småfeil må man likevel regne med.

Gangen i det nye REGARD-systemet er omtrent som følger.

1. MODAG-tall for bruttoprodukt, sysselsetting, brutto investering og arbeidsstyrke hentes inn. Disse splittes så opp i regionale tall. Disse operasjonene erstatter den gamle formodellen (Del II & III).
2. Yrkesprosentene blir regnet ut ved hjelp av tall hentet fra MOSART (Del III).
3. Hoveddelen av REGARD, modellen ARBEID, kjøres så det spesifiserte antall år. Man kjører kun ett år av gangen. Mellom hver 'runde' utføres demografiske oppdateringer mot *ettårige aldersgrupper*, slik som før (se Mohn [1994]) (Del IV & V).
4. Til slutt eksporteres data. Resultatdata for befolkning, netto flytting, arbeidsstyrke, sysselsetting og markedsleie blir kopiert til store matriser på rent tekstformat. Disse kan så eksporteres for eksempel til SAS, for videre analyse (Del VI).

Før modellen kan kjøres må vi sørge for at datagrunnlaget og modellen er etablert. For det første må data hentes inn fra det fylkesvise nasjonalregnskapet (FNR). Disse tallene brukes til å beregne andeler, som gjør oss i stand til å splitte nasjonale tall opp i regionale verdier (hvordan dette gjøres står i del II). Denne operasjonen trenger man bare gjøre hver gang det kommer nye FNR-tall, noe som er relativt sjelden.

Hver gang modellen skal kjøres må vi også sørge for at det er riktige MODAG(KVARTS)-tall som hentes inn. Dette gjøres ved å kjøre macroen *laggrlag*. Hver gang denne macroen kjøres blir vi bedt om å oppgi hvilke(n) database(r) med (MODAG/KVARTS)-tall som skal ligge til grunn for kjøringen. Macroen henter så inn de nødvendige data, kopierer dem til en mindre lokal database, *formod.db*, og gjør

eventuelle KVARTS-tall om til årlige verdier. Før vi kan bruke disse tallene i REGARD-simuleringer må vi utføre eventuelle 'manuelle' justeringer. Mer om dette i del II.

DEL I

Katalogstruktur og Filoversikt

1 Katalogstruktur

Alle filene som inngår i REGARD ligger på underkataloger av følgende katalog på frisch:

`/ssb/frisch/fame/regard`

Når vi senere i dette notatet refererer til en gitt underkatalog, menes alltid en av følgende underkataloger av dette området:

- **regard** - Her ligger alle kjørbare filer (`.prg`), det vil si at modellen kjøres fra denne katalogen. Tekstfilene som produseres blir også liggende på denne katalogen. Her ligger også input-filene for å lese inn FNR-tall, samt de innleste FNR-tallene. Noen av hjelpematrixene til REGARD blir også lagret på denne katalogen.
- **grlag** - Her ligger grunnlagsdata lagret: Historiske befolkningsdata, fremskrevne demografiske data, fremskrevne eksogene størrelser, samt MODAG/KVARTS-data (som kopieres hit før man kjører REGARD). Matriser og vektorer lagres på TROLL-databaser (`.tr1-fil`), mens tidsserier lagres på FAME-databaser (`.db`).
- **data** - Her lagres arbeidsdatabasene, dvs de databasene som man opererer på når man kjører modellen. I utgangspunktet inneholder disse en kopi av grunnlagsdata, men de oppdateres underveis. Når kjøringen er ferdig inneholder arbeidsdatabasene resultatdata.
- **koeff** - Inneholder databasen med koeffisientene til likningene i modellen ARBEID.
- **res** - Denne underkatalogen inneholder diverse matriser og mellomregninger produsert av eksportdelen av REGARD.
- **macro** - inneholder alle kildefilene (`.src`) til macroene i REGARD, både styringsmacroer, macroene til formodellen, macroer for oppdatering av demografiske størrelser, diverse hjelpemacroer, og macroene som genererer likningene i modellen ARBEID.
- **prodinnt** - Her ligger alt som tilhører den gamle modellen PRODINNT. Macroene ligger på underkatalogen **macro** av denne igjen, mens data, og inputfiler som genererer data ligger på underkatalogen **data**.
- **ettaar** - Her ligger macroene (kildekoden) til den foreløpige ettårige versjonen av REGARD.
- **dok** - inneholder den (tekniske) dokumentasjonen til systemet, først og fremst dette notatet, som er skrevet i L^AT_EX2 ϵ .

2 Filoversikt

Her følger en oversikt over alle filene som inngår i REGARD.

2.1 Data og koeffisienter

Alle data som inngår i REGARD ligger enten lagret på FAME-databaser (.db-filer), TROLL-databaser (.trl-filer), såkalte FORMDATA-filer (.frm-filer), eller som rene TROLL-datafiler (.dat-filer). Alle grunnlagsdata, arbeidsdatabaser, og koeffisienter er lagret på FAME- eller TROLL-databaser. Det er kun 'midlertidige' matriser, vektorer og tidsserier (mellomregninger) som lagres direkte som .dat-filer. Formdatafiler ble tidligere brukt blant annet når man overførte data fra gamle TROLL i Norges Bank, men er nå unødvendig.

Nasjonale tall fra MODAG/KVARTS

De nasjonale tallene som hentes fra MODAG eller KVARTS, og som splittes opp i regionale størrelser i formodellen, er lagret på FAME-databaser. Disse databasene blir kopiert fra deres opprinnelige område til (under)katalogen `grlag`. Siden det er kun få data fra disse store MODAG- og KVARTS-databasene vi har behov for i REGARD, har vi laget en liten macro, `laggrlag`, som henter ut de seriene vi har behov for (bruttoprodukt, investering, sysselsetting, arbeidsstyrke, antall utlendinger i utenriks sjøfart, antall vernepliktige i alder 20-24 år), og kopierer disse til en relativt liten (og lokal) FAME-database med navn `formod.db`. Det er denne lille databasen som blir aksessert, og hvor data hentes fra når vi kjører REGARD. Tall som hentes fra KVARTS har kvartalsvis periodisitet. Disse må derfor gjøres om til årlige tall. Dette blir også gjort i macroen `laggrlag` ved hjelp av TROLL-kommandoen `compact`.

Inntil videre må også de store databasene med MODAG- og KVARTS-tall kopieres til underkatalogen `grlag`, før de nødvendige dataene hentes ut og kopieres til den lille databasen `formod.db`. På sikt ønsker vi å kunne kopiere MODAG- og KVARTS-tallene direkte fra deres opprinnelige beliggenhet. Dette krever at REGARD-gruppens medlemmer får (lese)tilgang både på den katalogen hvor databasene befinner seg, samt databasene selv. I så fall må macroen `laggrlag` endres i henhold til dette, dvs setningen som aksesserer databasene må endres slik at man aksesserer riktig katalog.

For antall vernepliktige i alderen 16-19 år har vi MODAG-tall kun til og med 1995. Disse tallene ble tidligere hentet fra databasen `matauk.db`. Vi har nå kopiert denne serien til databasen `formod.db`, og forlenget den under antagelsen at denne gruppen holder seg konstant etter 1995. I og med at denne gruppen påvirker den totale arbeidsstyrken lite, vil ikke en slik antagelse få betydelige konsekvenser.

Historiske befolkningstall

For perioden fra og med 1991 til og med 1996 erstattes de fremskrevne befolkningstallene med historiske (reelle) tall. Disse er overført fra COMPAREX stor-maskin, og blir lagret som store matriser på TROLL-databasen `historie.trl` på katalogen `grlag`. For befolkningen etter utdanning har vi kun tall til og med 1995.

Data fra gamle TROLL

(Fremskrevne) eksogene (og endogene) data ble opprinnelig overført fra gamle TROLL i Norges Bank. Disse ble først lagret på såkalte FORMDATA-filer, og følgende filer inngikk:

- `formod.frm` - Herfra ble variabelen `ls60u` (antall sysselsatte utlendinger i utenriks sjøfart) hentet.
- `dgraf.frm` - Herfra hentet man inn eksogene størrelser som brukes i demografi-delen av REGARD, blant annet `bef`, `fbk`, `fbm`, `flb`, `fv`, `gu`, `sk`, `sm`.
- `arbeid.frm` - Inneholder historiske og/eller fremskrevne tall for eksogene og endogene størrelser til hovedmodellen ARBEID: `as`, `barn`, `bef`, `bm`, `butg`, `b`, `flrb`, `gamma`, `inb`, `inv`, `in`, `jb`, `jnp`, `ls`, `l`, `mgam`, `np`, `s.as`, `s.inb`, `s.in`, `s.np`, `s.utb`, `s.ut`, `syss`, `utb`, `utv`, `ut`, `xml`, `yp`.
- `mosart.frm` - Henter variablene `m` og `y2m`, som opprinnelig er beregnet i MOSART.

Alle de ovenfor nevnte FORMDATA-filene er kopiert til databaser med grunnlagsdata på katalogen `grlag`. Navnene på disse er henholdsvis `grlag.db` og `grlag.tr1`. Det første som skjer når vi starter en REGARD-kjøring er at disse grunnlagsdata blir kopiert til de såkalte *arbeidsdatabasene* ved hjelp av macroen `kopi`. Tidsserier kopieres til en FAME-database (`.db`), mens matriser og vektorer lagres på en TROLL-database (`.tr1`). Navnene på disse databasene bestemmes av navnet på kjøringen som brukeren oppgir. Hvis brukeren for eksempel oppgir `reg1997`, vil de to databasene få navn henholdsvis `reg1997.db` og `reg1997.tr1`. Arbeidsdatabasene er viktige også fordi resultatene av kjøringene (samt noen mellomregninger) havner på disse.

I dag består altså databasene `grlag.db` og `grlag.tr1` av en kombinasjon av historiske og fremskrevne tall beregnet av gamle TROLL. Vi har derfor ingen garanti for kvaliteten på disse dataene. En nødvendig forutsetning for at REGARD i fremtiden skal kunne produsere pålitelige resultater, er at man tar en grundig gjennomgang av alle grunnlagsdata, og eventuelt erstatter de fremskrevne data man har hentet fra gamle TROLL med nye og 'riktigere' data.

Koeffisienter

Koeffisientene (i modellen ARBEID) ble tidligere også hentet fra gamle TROLL, og de ligger som nevnt lagret på FAME-databasen `koeff.tr1`. Kopieringen av koeffisientene fra gamle TROLL til denne FAME-databasen ble gjort av input-fila `koeff.inp`. Som nevnt i innledningen fins det et relativt detaljert opplegg i SAS for å beregne nye koeffisienter. På sikt ønsker vi å gjøre dette opplegget mer strømlinjeformet og enkelt å bruke, selv om dette er en operasjon som ikke gjøres særlig ofte (bare hver gang selve modellen skal endres). Dagens opplegg for oppdatering av koeffisienter faller utenfor selve REGARD-modellen, og er derfor ikke nærmere dokumentert her. Faktisk fins det pr. i dag ikke noen fullgod dokumentasjon av dette opplegget. Vi henviser til Lasse Stambøl for nærmere informasjon om oppdatering av koeffisienter.

FNR-tall

Data fra Fylkesvist nasjonalregnskap (FNR-data) leses inn som store matriser på rent tekstformat. Disse filene overføres fra PC til UNIX (for eksempel ved hjelp av overføringsprogrammet `ftp`). Fra disse store matrisene lages input-filene `brprod.inp`, `invest.inp` og `syss.inp`, som blant annet inneholder et kall på nyttemacroen

medit. Når disse input-filene kjøres i TROLL, lages (TROLL-)matriser som inneholder FNR-tallene. Mer om dette i del II.

Hjelpematriser

I REGARD, og da spesielt i demografidelen, brukes en rekke hjelpematriser i de matriseoperasjonene som utføres. Noen av disse lagres på grunnlagsdatabasen `grlag.tr1`, mens andre lagres som `.dat`-filer (TROLL-matriser) på den (under)katalogen man kjører modellen fra; `regard`. I sistnevnte kategori hører de tre matrisene `cc`, `cu` og `cbarn` hjemme. Dette er hjelpematriser for oppsplitting/aggregring mellom ettårig og flerårig aldersstruktur.

2.2 Macroer

Filer som inneholder kildekode til macroer har alltid ekstensjon `.src`, mens ferdig kompilerte (kjørbare) versjoner har ekstensjon `.prg`. Når macroer skal kjøres trenger man ikke å oppgi ekstensjonen, men man må istedet sette et `&`-tegn foran, på følgende måte: `&formod`; Her følger kun en grov oversikt over hvilke macroer som inngår i systemet, og hvilke oppgaver de utfører. I de etterfølgende kapitlene gis mer detaljerte beskrivelser av funksjonsmåte, eventuelle parametre etc.

Nyttmacroer

- `medit` - Leser inn (store) matriser lagret på rent tekstformat, og genererer en Troll-matrise (`.dat`-fil). Brukes til innlesing av FNR-tall.
- `kombiner` - Macro som virker som `partcomb`-kommandoen i (gamle) TROLL. Trengs når man kjører på lavere versjonsnummer enn 1.0 i POTROLL. Er nå i all hovedsak erstattet av den nye versjonen av `partcomb`.
- `colsplit` - Virker som `colsplit`-kommandoen i gamle TROLL.
- `lagmat` - Lager aggregeringsmatriser ved hjelp av spesifikasjonen gitt av brukeren. Disse matrisene består av enere og nuller, og aggregeringen utføres ved å matrisemultiplisere disse med de uaggregererte dataene. I REGARD brukes dette først og fremst til å aggregere FNR-tall til MODAG-nivå i forbindelse med beregning av andeler i formodellen. Det fins også en egen versjon av denne macroen (`lagmat2`) som lager aggregeringsmatriser for aggregering fra ettårige til flerårige aldersgrupper.
- `kopi` - Kopierer grunnlagsdata over i arbeidsdatabasene. Det fins en versjon som kopierer FORMDATA-filene med historiske data (hentet fra gamle TROLL), og lagrer de på databasene med grunnlagsdata. Kalles av styringsmacroen `regard`.
- `laggrlag` - Kopierer nødvendige data fra MODAG og/eller KVARTS-databasene med nasjonale tall til databasen `formod.db`. Gjør KVARTS-tallene om til årlige verdier.
- `res` - Beregner vekstprosenten mellom to gitte årstall for variable produsert av formodellen, dvs bruttoprodukt, investering og sysselsetting etter sektor og fylke (region).

- `laghjelpmat` - Lager de tre hjelpematrixene `cc`, `cu` og `cbarn`, og lagrer disse som TROLL-filer (`.dat`) på underkatalogen `regard`, der systemet kjøres fra. Disse matrixene brukes i demografidelen til å splitte opp demografiske data fra REGARD's flerårige aldersgruppeinndeling til ettårig inndeling, samt å aggregere tilbake. Kalles av macroen `regard`.
- `aggcc` - Denne macroen kalles av `laghjelpmat`, og lager matrixen `cc`. Kaller igjen på macroen `lagmat2` med spesifikasjonene for aggregering fra ettårige til flerårige aldersgrupper som parameter. Dette produserer matrixer bestående av enere og nuller, og som brukes til oppsplitting/aggregering.
- `aggcu` - Lager hjelpematrixen `cu` på tilsvarende måte som `aggcc`.
- `aggcbarn` - Lager hjelpematrixen `cbarn` på samme måte som de to foregående macroene.

Formodellen

- `formod` - Styringsmacro for formodellen. Utfører kall til macroer som aggregerer FNR-tall til MODAG-nivå og beregner andeler, som beregner (ekso-gene) fylkesfordelte tall for investering, bruttoprodukt, og sysselsetting. Macroen `formod` avsluttes med at de beregnede tallene aggregeres fra fylke til region.
- `fylkvis` - Aggregerer fra NR-nivå til MODAG-nivå, og beregner de regionale (fylkesvise) andelene for bruttoprodukt, investering og sysselsetting.
- `afratilps` - Lager aggregeringsmatrixen for bruttoprodukt (ved å kalle på hjelpemacroen `lagmat`).
- `afratiljs` - Lager aggregeringsmatrixen for investering.
- `afratilsy` - Lager aggregeringsmatrixen for sysselsetting.
- `invest2` - Beregner fylkesvis investering på MODAG-nivå, etter sektor.
- `lagqseq` - Beregner fylkesvist bruttoprodukt etter sektor.
- `syss` - Beregner fylkesvis sysselsetting etter sektor.
- `arbstyrk` - Henter inn tall for total arbeidsstyrke fra MODAG/KVARTS. Korrigerer med vernepliktige og utlendinger i utenriks sjøfart.
- `yprmod2` - Henter inn tall fra MOSART, og beregner yrkesprosenten.

Generering av likninger

- `arbeid` - Definerer ARBEID som 'aktuell' modell (ved hjelp av TROLL-kommandoen `usemod`), og kaller på de likningsgenererende macroene.
- `nysyssutd` - Lager likninger for arbeidsstedets sysselsetting.
- `gam2` - Skalerer sysselsettingen slik at den er lik uansett hvordan den splittes opp.
- `ettersp2` - Lager likninger for pendling og markedsleie.

- nyutd - Lager likninger for oppdatering av befolkningstall.
- flyttmod - Lager likninger for utflytting.
- flytmod2 - Lager likninger for innflytting.
- flytmod5 - Lager likninger for flytting av barn.
- innutv - Lager likninger for inn- og utvandring.
- nysyss - Lager likninger for sysselsetting etter bosted.
- nytilbud - Lager likninger for arbeidsstyrken.

Kjøring av REGARD, demografimacroer

- regard - Styringsmacro for hele REGARD-modellen. Utfører nødvendige søk, kaller på formodellen, og starter simuleringen.
- dgraf - Styringsmacro for oppdatering av demografiske størrelser.
- ninv - Fremskriver inn- og utvandring for befolkningene i regionene. Oppdaterer befolkningen med netto innvandring.
- flytt - Justerer ettårige befolkningsmatriser med netto innenlandsk flytting.
- newborn - Legger fødselstall til befolkningen. Aldrer befolkningen.
- innbef - Aggregerer befolkningen til REGARD's aldersgrupper.
- rb - Beregning av dødelighet etter ettårig alder.
- utbef - Aggregerer den foreløpige demografiske fremskrivningen.

Eksport av resultater

- export - Styringsmacro, som kaller på de macroer som eksporterer resultatene til tekstfiler.
- exbef - Eksporterer befolknings- og flyttetall (til tekstfiler).
- arbst - Eksporterer tall for arbeidsstyrke.
- sysjr - Eksporterer sysselsettingstall.
- xml - Eksporterer tallene for markedsløien.

2.3 Resultat-filer

Resultater av REGARD-kjøringer (som i utgangspunktet er lagret som tidsserier og matriser på arbeidsdatabasene), blir avslutningsvis satt sammen på tabellform, og kopiert til (store) tekstfiler (.txt). Hver enkelt rad i disse filene inneholder kjennetegn som årstall, kjønn, utdanningsnivå og alder. Deretter følger sju verdier som angir verdien av den aktuelle variabelen i de sju regionene. Det er 13 slike tekstfiler, med tall for befolkning totalt (**bftot**), befolkning etter utdanning (**befutd**), flytting totalt (**nfbtot**), flytting etter utdanning (**nfbutd**), arbeidsstyrke etter utdanning (**arbst**), (bosteds-)sysselsetting (**sysjr**), og til slutt markesleien (**xml**). Disse filene er lokalisert der hvor REGARD kjøres fra, underkatalogen *regard*. Se mer om dette i del VI.

DEL II

Innhenting av data. Styringsmacroer

1 Innhenting av FNR-tall

Før en REGARD-kjøring kan startes må vi som nevnt på forhånd ha laget andelene som splitter MODAG- (eller KVARTS-)tallene opp i regions- eller fylkestall. Disse andelene blir beregnet ved hjelp av tall fra det fylkesvise nasjonalregnskapet (FNR). Denne operasjonen behøver vi bare utføre hver gang det kommer nye FNR-tall, noe som skjer relativt sjelden. FNR-tallene blir overført fra nasjonalregnskapet på rent tekstformat. Tallene blir først kopiert til filer på PC. Deretter overføres de til UNIX, for eksempel ved hjelp av overføringsprogrammet `ftp`. Filene blir overført til underkatalogen `regard`. REGARD er nå forenklet slik at vi kun trenger FNR-tallene for bruttoprodukt, brutto investering, og sysselsetting. FNR-tallene ligger lagret som store matriser, med NR-kodene loddrett, og fylkeskodene vannrett. Før tallene kan leses inn og brukes av REGARD, må både NR- og fylkeskodene i hode og forspalte fjernes. Den siste kolonnen, og den nederste raden, som inneholder henholdsvis rad- og kolonnesummene, fjernes også. Brukeren må så legge inn følgende linje *først* i hver av de tre filene med FNR-tall:

```
&medit navn x y
```

der *navn* enten er `invest`, `brprod` eller `syss`, avhengig av hvilken matrise det dreier seg om (investering, bruttoprodukt eller sysselsetting). Parameteren *x* og *y* er tall som forteller henholdsvis hvor mange rader og kolonner det er i den korresponderende matrisa. Antall kolonner, *y*, er lik antall fylkeskoder, dvs normalt 20 (inkludert et ekstrasfylke for virksomhet utenfor fastlands-Norge). Antall rader i matrisen er antall NR-koder som inngår, normalt mellom 160 og 180. Etter det siste elementet i hver av filene må man legge inn et semikolon. Disse tre ferdig editerte datafilene lagres på *inputfiler* med navn henholdsvis `brprod.inp`, `invest.inp` og `syss.inp`.

TROLL startes opp, og de tre nevnte inputfilene kjøres. Dermed lages tre TROLL-matriser av de editerte FNR-dataene: `brprod.dat`, `invest.dat` og `syss.dat`. Vi har laget en liten input-fil `fnr.inp`, som søker på underkatalogen `regard`, og som utfører operasjonen over. Skjematisk ser altså overføringen av FNR-tall ut som følger:

1. Kopier de tre (tekst)filene med FNR-tall for bruttoprodukt, investering og sysselsetting fra nasjonalregnskapet til en passende katalog på din egen PC.
2. Overfør filene til underkatalogen `regard` på UNIX (ved hjelp av `ftp`).
3. Editér (ved hjelp av en passende tekst-editor, f.eks. `emacs`) disse tre filene som beskrevet over, ved å legge inn kall på macroen `medit`, og fjern hode, forspalte, samt rad- og kolonnesummene. Lagre de editerte filene på *input-filer* som beskrevet over. Alternativt kunne man laget disse *input-filene* ved hjelp av en editor på PC (f.eks. `Qedit` eller `TSE`), og så overført de ferdig editerte *input-filene* til UNIX.
4. Start TROLL, og kjør *input-filen* `fnr.inp` (skriv `input fnr` i TROLL). Resultatet er som beskrevet over tre matriser i TROLL (på `.dat`-format), som inneholder FNR-tallene. Disse matrisene blir i henhold til oppgitt søk liggende på underkatalogen `regard`.

I formodellen blir matrisene `brprod.dat`, `invest.dat` og `syss.dat` aggregert til MODAG-nivå, og blir brukt til å beregne andelene for å splitte opp nasjonale tall

fra MODAG(KVARTS) til fylkes- og region-nivå.

2 Tall fra MODAG/KVARTS

Macroen `laggrlag` henter som nevnt ut de seriene vi har behov for fra databasen(e) med MODAG(KVARTS)-tall, og kopierer disse til en (mindre) database, `formod.db`. I den nye versjonen av REGARD trenger vi som tidligere nevnt kun seriene med investering, bruttoprodukt, sysselsetting, arbeidsstyrke, antall vernepliktige, samt utlendinger i utenriks sjøfart. Resten av innholdet i de svært store databasene med MODAG/KVARTS-tall har vi ikke behov for. Når macroen `laggrlag` kalles, blir brukeren bedt om å oppgi hvorvidt det er MODAG- eller KVARTS-tall som skal ligge til grunn for kjøringen, ved å taste enten 'M' eller 'K'. Deretter blir brukeren bedt om å oppgi navnet på den MODAG (eller KVARTS) databasen vi skal hente tall fra. I tilfellet at vi bruker tall fra MODAG er opplegget helt enkelt. De ønskede seriene kopieres til vår lille database `formod.db` på samme måte som i macroen `kopi`. I tilfellet at en KVARTS-kjøring skal brukes, er opplegget noe mer komplisert. Brukeren blir da bedt om å oppgi navnet på den KVARTS-databasen som skal brukes. I tillegg blir han eller hun bedt om å oppgi navnet på den nyeste databasen med MODAG-tall som er tilgjengelig. Hvorfor skal vi komme tilbake til. Når vi skal kopiere KVARTS-tallene til databasen `formod.db` må de først gjøres om til *årlige tall*. Dette gjøres med TROLL-kommandoen `compact`. En typisk kopiering kan se ut som følger:

```
>>dofile ut_jks&(se) = compact(in_jks&(se),-1,1);
```

der `ut` er aliaset til databasen `formod.db`, mens `in` er aliaset til KVARTS-databasen. Parameteren '-1' betyr at de fire kvartalsvise verdiene i KVARTS-databasen skal adderes så vi får en årlig verdi. For noen variable vil det riktige være å bruke gjennomsnittet av de fire kvartalsvise verdiene. I de tilfellene gis parameteren '0' istedet. Den siste parameteren angir periodisiteten på vår 'nye' serie ('1' betyr årlig periodisitet). Se mer om `compact`-kommandoen i TROLL-manualen. Et problem med å bruke KVARTS-tall i REGARD, er at vi mangler sysselsettingstall for selvstendig næringsdrivende *etter sektor* i KVARTS. Dermed må slike tall estimeres. Til dette trenger vi MODAG-databasen. Vi splitter opp totaltallet for sysselsetting for selvstendig næringsdrivende fra KVARTS slik at *fordelingen* på de ulike sektorene stemmer overens med sysselsetting for selvstendig næringsdrivende etter sektor i MODAG. TROLL-kommandoen for å gjøre dette ser omtrent ut som:

```
>>dofile ut_ns&(se) = (ut_ns/nsmod)*in_ns&(se);
```

der `ut` er aliaset til `formod.db`, og `in` er aliaset til MODAG-databasen, `ut_ns` er total sysselsetting for selvstendig næringsdrivende hentet fra KVARTS (og gjort om til årlige tall), mens `nsmod` er den totale sysselsettingen for selvstendige næringsdrivende hentet fra MODAG.

2.1 Eventuelle 'manuelle' justeringer

I noen spesielle tilfeller kan ikke ujusterte tall hentet fra databaser produsert av modellkjøring (i MODAG/KVARTS) brukes direkte som de er. I slike tilfeller må vi selv justere tallene på databasen `formod.db` før vi kan starte REGARD-kjøringen.

Detaljene for hvordan dette skal gjøres vil kunne variere noe fra gang til gang, delvis fordi det kan være forskjellige tidsserier som skal justeres, og delvis fordi måten disse seriene skal endres på kan variere noe. Derfor kan vi bare skissere hvordan dette generelt vil se ut her. Slike operasjoner krever at brukeren har god innsikt i hvordan seriene skal justeres, og hvordan slike endringer foretas (i TROLL). Alternativt kan slike manuelle eller *interaktive* justeringer av databasen `formod.db` utføres direkte i FAME, for den som måtte foretrekke det. Her skal vi kun se på hvordan dette kan gjøres i TROLL. Vi antar at brukeren befinner seg på underkatalogen `regard`, der hele systemet jo kjøres fra.

Først må databasen `formod.db` aksessereres. Dette gjøres ved å starte TROLL, og skrive kommandoen (merk at vi her ikke har med dobbelt 'større-enn' tegn, `>>` foran kommandoen. Som kildekode i en TROLL-macro må vi alltid ha med `'>>'` foran TROLL-kommandoer. Når man kjører TROLL interaktivt behøves ikke dette):

```
access inn type fame id /ssb/frisch/fame/regard/grlag/formod.db mode rw;
```

Deretter kan vi endre tidsserier ved å gi passende TROLL-kommandoer. Dette skisseres best ved eksempler. Dersom vi skal erstatte en gitt serie, for eksempel arbeidsstyrken med konkrete tall, kan vi utnytte `combine-` og `reshape-`kommandoen i TROLL:

```
dofile inn.nt = reshape(combine(2000,2010,2020,2030,...,2100),1992A);
```

der altså 2000,...,2100 er verdiene serien skal ha, og parameteren 1992A forteller at serien skal starte i 1992 og ha årlig periodisitet. Dersom kun en del av en lengre tidsserie skal endres, kan vi alternativt bruke `overlay-`funksjonen i TROLL. For eksempel, om vi skal erstatte observasjonene fra 1996 til 1998 med nye verdier, kan dette gjøres på følgende måte:

```
dofile inn.nt = overlay(reshape(combine(2010,2020,2030),1996A),inn.nt);
```

Legg merke til at `overlay-`kommandoen overskriver *fra venstre*, dvs den første serien overskriver eventuelle etterfølgende, der seriene har felles observasjoner. Merk også at seriene må ha lik periodisitet. Dersom en av seriene, for eksempel sysselsetting for arbeidstakere, `nw`, skal endres slik at vi har en årlig vekstprosent istedet for de allerede innleste tall, kan dette gjøres ved hjelp av `autocum-`funksjonen i TROLL:

```
dofile inn.nw = autocum(inn.nw,0,1992A,0,1.02);
```

Dette innebærer at den opprinnelige serien fra og med 1992 erstattes av en ny serie som består av 1992-nivået til den opprinnelige serien fremskrevet med en årlig vekstprosent på 2 prosent, eller ekvivalent, med en faktor 1.02. For mer detaljer omkring interaktiv bruk av TROLL, og hvordan de ulike TROLL-kommandoene fungerer og brukes, henvises til TROLL-manualen (Hollinger og Spivakovski [1996]).

3 Styringsmacroen `regard`

REGARD kjøres ved å kalle på styringsmacroen `regard` fra TROLL. Denne macroen har tre parametre som kan leses inn fra terminal. Alternativt kan kallet på macroen `regard` legges inn på en inputfil sammen med parameterverdiene. Et kall på macroen kan for eksempel se slik ut:

®ard tull 1992A 2000A;

Den første parameteren (tull) er navnet på kjøringen. De to andre er henholdsvis første og siste simuleringsår. Navnet på kjøringen brukes til å opprette to utdatabaser (arbeidsdatabasene) med angitt navn. Den ene er som tidligere nevnt en FAME-database hvor tidsserier lagres, den andre en TROLL-database hvor matriser og vektorer legges. Disse databasene aksesseres med følgende setninger:

```
>>access ut1 type fame id /ssb/frisch/fame/regard/data/&(navn).db
>>mode rw;
>>access ut2 type trolldb id /ssb/frisch/fame/regard/data/&(navn).trl
>>mode rw;
```

Når modellen kjøres, aksesseres databasene med modus *update* (mode u) istedenfor 'rw' fordi databasene på det stadiet allerede er opprettet. Vi ønsker kun å oppdatere dataene framfor å overskrive seriene. Setningene over sikrer at hver kjøring får sin egen resultatfil, og det blir derfor enkelt å holde styr på de ulike kjøringene.

Strukturen i macroen `regard` er omtrent som antydnet i innledningen. Først kopieres grunnlagsdatabasene til arbeidsdatabasene ved hjelp av macroen `kopi`. Deretter lages hjelpematrixene `cc`, `cu` og `cbarn` ved hjelp av macroen `laghjelpmat`. Arbeidsdatabasen blir så aksessert. Når dette er utført kjøres formodellen (ved hjelp av macroen `formod`), som leser inn MODAG-tall, og utfører de nødvendige aggregeringer og oppsplittinger. Yrkesprosentene blir beregnet med et kall på macroen `yprmod2`. Selve modellkjøringen utføres i en `for`-løkke, hvis lengde er bestemt av parametrene for første og siste simuleringsår. For hver runde i denne løkka simuleres modellen ARBEID for ett år av gangen supplert med oppdatering av demografiske data. Den siste operasjonen utføres med et kall på macroen `dgraf`. Når hele simuleringen er fullført lagres de beregnede tallene på ut-databasene (arbeidsdatabasene). Helt til slutt kopieres (de viktigste) resultatene av simuleringen til store tekstfiler ved hjelp av et kall på macroen `export` (som igjen kaller på en rekke undermacroer).

3.1 Hjelpemacroen kopi

Macroen `kopi` kopierer som nevnt grunnlagsdatabasene (databasene med fremskrevne demografiske størrelser og eksogene variable til modellen) `grlag.db` og `grlag.trl` til arbeidsdatabasene, med navn oppgitt av den som kjører modellen. Strukturen i macroen er som følger:

```
>>access inn type trolldb id /ssb/frisch/fame/regard/grlag/grlag.trl
>>mode r;
>>search data inn;
>>access ut2 type trolldb id /ssb/frisch/fame/regard/data/&(navn).trl
>>mode rw;
>>dofile ut2_bef1997 = bef1997;
```

Setningene over sørger for at matrisen `bef1997` som ligger på TROLL-databasen `grlag.trl` blir kopiert til arbeids- og resultatdatabasen `&(navn).trl`, der altså `navn` er variabelen som inneholder navnet på kjøringen (som oppgitt av brukeren). Macroen `kopi` inneholder derfor en rekke TROLL-setninger av typen

```
>>dofile ut2_x = x;
```

der `x` er navn på variable (matriser, vektorer eller tidsserier) i grunnlagsdatabasene.

Dermed blir innholdet av grunnlagsdatabasene kopiert matrise for matrise, eller serie for serie, til arbeidsdatabasene, identifisert med aliaset ut2 (eller ut1).

Hjelpemacroen laghjelpmat, som også blir kalt opp fra regard, produserer som nevnt de tre matrisene cc, cu og cbarn. Disse matrisene lagres som TROLL-matriser på underkatalogen regard, og brukes til å splitte opp fra flerårige til ettårige aldersgrupper samt aggregere tilbake i demografidelen. Derfor henviser vi til dokumentasjonen av demografimodulen for en mer detaljert beskrivelse av denne macroen.

DEL III

Formodellen. Yrkesprosenter

1 Innledning

Formodellen er ingen egentlig *modell*, men består av en rekke *eksogene* utregninger, som beregner tall for bruttoprodukt, investering og sysselsetting, oppsplittet etter region og sektor. Denne modulen kjøres ved at styringsmacroen `regard` kaller på macroen `formod`. Dersom det er ønskelig er det ingenting i veien for å kjøre formodellen separat, ved å kalle på `formod` direkte fra TROLL. Dette forutsetter imidlertid at man har gjort alle forberedelser med FNR-tall, MODAG-tall etc. som ordnes av macroen `regard`.

2 Variable i Formodellen

FNR-tallene som trengs i formodellen er lagret på følgende tre Troll-matriser (se del II om innlesing av FNR-tall):

- `brprod.dat` - Bruttoprodukt
- `invest.dat` - Investering
- `syss.dat` - Sysselsetting

Disse ligger på underkatalogen `regard`, og brukes som input i macroen `fylkvis`. De tilsvarende matrisene aggregert til MODAG-nivå lagres på TROLL matrisene `bruttprod.dat`, `investering.dat` og `syssel.dat`. Andelene for å beregne fylkesvist fordelt bruttoprodukt, investering og sysselsetting på MODAG-nivå beregnes også i macroen `fylkvis`. Disse lagres på Troll-vektorene

- `fqs.dat` - Andel bruttoprodukt.
- `fjs.dat` - Andel investering.
- `fls.dat` - Andel sysselsetting.

Disse ligger også på underkatalogen `regard`. I formodellen leses følgende tall inn fra MODAG (KVARTS) til fylkesvis oppsplitting:

- `q` - Bruttoprodukt etter sektor. Brukes i macroen `lagqseq`.
- `jks` - Investering etter sektor. Brukes i macroen `invest2`.
- `n` - Sysselsetting etter sektor. Brukes i macroene `syss` og `arbstyrk`.

Følgende variable hentes inn fra MODAG (KVARTS) for å beregne den totale arbeidsstyrken (i macroen `arbstyrk`) slik den skal inngå i hovedmodellen ARBEID:

- `nt` - Total arbeidsstyrke.
- `ls60` - Antall utlendinger i utenriks sjøfart.
- `nvpl16` - Andelen vernepliktige i alder 16-19 år.
- `nvpl20` - Andelen vernepliktige i alder 20-24 år.

Output fra formodellen er tallene for bruttoprodukt, investering og sysselsetting fordelt etter fylke og sektor. Disse størrelsene blir også aggregert til REGARD's sju regioner i macroen `formod` (for en sammenheng mellom oppsplittingen i fylker og REGARDS sju regioner henviser vi til appendiks A). De regionfordelte tallene for sysselsetting brukes direkte i 'hovedmodellen' ARBEID.

- `qs` - Brutttoprodukt etter fylke og sektor. Fylkesvise tall beregnet i `lagqseq`.
- `js` - Investering etter fylke og sektor. Fylkesvise tall beregnet i `invest2`.
- `ls` - Sysselsetting etter fylke og sektor. Fylkesvise tall beregnet i `syss`.

Output fra macroen `arbstyrk` er total arbeidsstyrke, lagret i variabelen `antot`. Denne variabelen inneholder total arbeidsstyrke fra MODAG minus antall vernepliktige og antall utlendinger i utenriks sjøfart.

3 Macroene i Formodellen

Vi har i del I gitt en oversikt over macroene som inngår i formodellen. Vi skal her studere macroene mer i detalj. Formodellen startes ved et kall på styringsmacroen `formod`. Denne macroen utfører først et kall på macroen `fylkvis`, som aggregerer FNR-tall til MODAG-nivå og beregner andeler for oppsplitting av MODAG/KVARTS-tall. Deretter beregnes (eksogene) fylkesfordelte tall for investering, bruttoprodukt, og sysselsetting, med kall på macroene `invest2`, `lagqseq` og `syss`. Arbeidsstyrken beregnes med et kall på macroen `arbstyrk`. Macroen `formod` avsluttes med at fylkesfordelte tall for bruttoprodukt, investering og sysselsetting blir aggregert til regionsnivå. Etter kallet på `formod`, kjøres macroen `yprmod2`, som henter tall fra MOSART, og beregner yrkesprosenter.

3.1 Macroen fylkvis

Det første som skjer i formodellen er at macroen `fylkvis` blir kalt. Denne macroen tar tak i FNR-tallene, og aggregerer de opp til MODAG-nivå. Ved hjelp av disse beregnes så andeler, som forteller hvordan (nasjonale) tall (hentet fra MODAG) skal splittes opp etter region og sektor, før de tas i bruk i hovedmodellen, ARBEID.

Før selve aggregeringen, kalles de tre macroene `afratilps`, `afratiljs` og `afratilsy`. Disse macroene består av et kall på hjelpemacroen `lagmat` med den korresponderende aggregeringsspesifikasjonen som parameter, på følgende måte (for en fullstendig sammenheng mellom sektorene på NR- og MODAG-nivå henvises til appendiks A samt kildekoden til macroene `afratilps`, `afratiljs` og `afratilsy`):

```
&lagmat;
>>ps
>>11 = 22010 + 22015 + 23010 + 23014 *
>>12 = 23020 *
.
.
.
>>'95K'' = 25410 + 25453 + 25751 + 25900 + 25921 *
>>**
```

Parametrene til lagmat er for det første 'typen' (i eksempelet over ps, som står for 'produksjonssektor', og som brukes ved aggregering av tallene for bruttoprodukt). Deretter kommer hele spesifikasjonen av aggregeringen. Dette produserer de såkalte *aggregeringsmatrisene*. Disse matrisene har navn `aggps.dat`, `aggjs.dat` og `aggsy.dat`, ligger lagret på (under)katalogen `regard`, og består av 0'ere og 1'ere. Aggregeringen utføres ved å multiplisere FNR-matrisene med disse matrisene:

```
>>dofile bruttoprod = matmult(aggps,brprod);
>>dofile investering = matmult(aggjs,invest);
>>dofile syssel = matmult(aggsy,syss);
```

Disse kommandoene utføres også av macroen `fylkvis`. Merk at man før aggregering foretar en justering av NR-sektor 23611, utenriks sjøfart, i matrisen for sysselsetting, slik at utlendinger i denne næringen ikke telles med. Vi antar da at andelen utlendinger i utenriks sjøfart er den samme regionvis som på landsbasis, fordi disse tallene ikke er fordelt fylkesvis. Vi må for øvrig trekke i fra både vernepliktige og utlendinger i utenriks sjøfart fordi disse kategoriene ikke inngår i de filene som fordeler sysselsetting og arbeidsstyrke på kjønn og utdanning.

Når aggregering er utført beregnes andelene ved å dividere hvert element i den aggregerte matrisen med den korresponderende radsummen. Dette produserer de tre matrisene med andeler; `fqs.dat`, `fjs.dat` og `fls.dat`. I TROLL-syntaks ser operasjonen ut omtrent som følger:

```
>>do nevner = matmult(bruttoprod,x);
>>do nevner = nevner + transp(seq(&(nf))*0);
>>dofile fqs = bruttoprod/ if nevner == 0 then 1 else nevner;
```

der `x` er en kolonnevektor bestående av kun 1'tall, og med like mange elementer som det er kolonner i matrisen med aggregerte FNR-tall. Den første linja produserer derfor en kolonnevektor bestående av radsummene i de aggregerte FNR-matrisene. Når hvert element i FNR-matrisene divideres med den korresponderende radsummen, får vi de ønskede andelene. Setningene for å produsere andeler for investering og sysselsetting, `fjs` og `fls`, er omtrent identisk med setningene over.

NB! Når vi beregner andeler i sektorer hvor utlendinger i utenriks sjøfart inngår (NR-sektor 23611), må vi som nevnt foreta en justering. Grunnen til dette er at NR-sektor 23611 legges sammen med sektor 23112 til MODAG-sektor 65. I NR-sektor 23112 inngår ingen tall for utlendinger i utenriks sjøfart. Den aggregerte verdien (MODAG-sektor 65) kan derfor skrives

$$v_{65} = (1 - fac) \cdot f_{23611} + f_{23112}$$

der f_{23611} og f_{23112} er verdiene i FNR for henholdsvis NR-sektor 23611 og 23112. `fac` er andelen utlendinger i utenriks sjøfart. Hvis vi ikke hadde tatt hensyn til utlendingene i utenriks sjøfart, ser vi at vi ville fått gal andel når vi dividerte med totaltallene (landsbasis) for de gitte sektorene. For de vernepliktige består aggregering fra FNR til MODAG av kun en sektor, og derfor får man samme korreksjonsfaktor både i teller og nevner når andelen skal beregnes. I dette tilfellet kan vi derfor se bort fra korreksjonen. Resultatet blir likevel det samme!

3.2 Macroene invest2, lagqseq og syss

Etter at FNR-tallene er aggregert, og andelene er beregnet, brukes disse til å splitte opp innleste data (fra MODAG/KVARTS) etter fylke. Dette skjer i macroene invest2, lagqseq og syss (som beregner fylkesfordelte tall for henholdsvis investering, bruttoprodukt og sysselsetting). Disse macroene er svært like i struktur. De består av en dobbelt for-løkke (over sektor og fylke), hvor man for hver kombinasjon utfører en kommando på formen:

```
>>dofile js&(se)&(fy)= &(fj)*jks&(se);
```

der *se* er sektorkoden, *fy* er fylkeskoden, koeffisienten *fj* er den korresponderende (og nylig beregnede) andelen fra macroen fylkvis, og variabelen *jks* er tall hentet fra MODAG/KVARTS (fordelt over sektor). Variabelen *js* (indeksert over sektor og fylke) er dermed fylkesfordelte MODAG eller KVARTS-tall. Eksempelet over viser hvordan fylkesfordelte tall for investering blir beregnet. Bruttoproduktet og sysselsettingen blir beregnet på samme måte.

3.3 Macroen arbstyrk

Macroen *arbstyrk* henter inn totale tall for arbeidsstyrken fra MODAG, som lagres i variabelen *nt*. Denne blåses opp med en faktor på 1000, siden MODAG-tallene er i antall tusen. For å finne den endelige verdien for den totale arbeidsstyrken i REGARD trekkes utlendinger i utenriks sjøfart, og antall vernepliktige i fra. Den endelige verdien for total arbeidsstyrke lagres som tidligere nevnt i variabelen *antot*. **NB!** Antall vernepliktige i aldersgruppen 16–19 år er nå tatt ut av MODAG. Serien med disse tallene må derfor hentes fra et annet sted. Tidligere hentet vi disse tallene fra databasen *matauk.db*, men i denne basen fins kun tall frem t.o.m. 1995, og vi har derfor forlenget tidsserien ved å anta at verdiene i de etterfølgende år er lik verdien i 1995. Som tidligere nevnt har serien med antall vernepliktige i alder 16-19 år nå blitt kopiert til databasen *formod.db*, slik at vi slipper å aksessere (den store) databasen *matauk*.

3.4 Aggregering fra fylke til region

Når tallene for bruttoprodukt, investering, sysselsetting og arbeidsstyrke er hentet fra MODAG (KVARTS), og fordelt på fylker, avsluttes macroen *formod* med at tallene for bruttoprodukt, investering og sysselsetting også aggregeres til REGARDS inndeling i sju regioner, gitt av Mohn et.al [1994]. Denne aggregeringen er vi avhengig av i ARBEID, der det opereres på regionnivå istedenfor fylke.

NB! Av de størrelsene som beregnes i formodellen, er det kun sysselsettingen og den totale arbeidsstyrken som brukes videre i modellen ARBEID.

3.5 Beregning av yrkesprosent

Tidligere var beregning av yrkesprosentene en del av modellen (ARBEID). Disse har vi nå flyttet ut, siden de essensielt blir beregnet ved en rent 'eksogen' beregning uavhengig av de endogene variablene i ARBEID. Med utgangspunkt i yrkesprosenten for året *før* det første simuleringsåret, blir yrkesprosenten oppdatert ved hjelp av

MOSART-variabelen y_{2m} . Yrkesprosentene skal ha samme endring som variabelen y_{2m} , og endring av yrkesprosent kan derfor skrives som:

$$yp_{i,l,j}(t) = yp_{i,l,j}(t-1) + y_{2m_{i,l,j}}(t) - y_{2m_{i,l,j}}(t-1)$$

der $yp_{i,l,j}(t)$ betyr yrkesprosenten etter kjønn, alder og utdanning, i år t . Beregningen av yrkesprosenter utføres av macroen `yprmod2`. Historiske og tidligere fremskrevne verdier av yrkesprosenter er lagret i variabelen `yp`, mens de nye oppdaterte verdiene lagres i variabelen (tidsserien) `ypr`.

4 Analyse av resultater. Beregning av vekstprosenter

Tallene for bruttoprodukt, investering og sysselsetting produsert av formodellen lagres som nevnt på arbeidsdatabasen som tidsserier med navn henholdsvis `qs`, `js` og `ls`. Hver serie er indeksert etter hvilken sektor og hvilket fylke (region) innholdet av serien gjelder. For eksempel vil serien som inneholder bruttoprodukt i sektor 14 (fiskeoppdrett) og fylke 12 (Hordaland) hete `qs1412`. Observasjonene i de ulike seriene inneholder fylkes(eller region)fordelte tall med årlig periodisitet, og kan både være oppsplittede historiske tall og fremskrevne verdier. I analyseøyemed er det viktig å finne ut hvordan de ulike seriene varierer over tid. Det er derfor av stor interesse å bestemme *vekstprosentene*. Vi har laget en macro i TROLL, `res`, som gjør beregningen av vekstprosenter enklere for brukeren. Kildefilen (`.src`) til denne macroen ligger på underkatalogen `macro`, mens den kjørbare versjonen (`.prg`) ligger på underkatalogen `regard`. Denne macroen kjøres ved å gå til underkatalogen `regard`, starte TROLL, og kjøre macroen `res`, ved å taste `&res` etterfulgt av å trykke ENTER. Brukeren blir først bedt om å oppgi navnet på FAME-databasen hvor resultat-seriene er lagret (etterfulgt av ENTER). Deretter blir brukeren bedt om å oppgi et årstall, og kan da taste for eksempel 1997 etterfulgt av ENTER. Macroen beregner da vekstprosentene fra 1997 til 1998 for alle seriene for bruttoprodukt, investering og sysselsetting produsert av formodellen. Brukeren vil oppleve at skjermen 'ruller' mens resultatene skrives ut. Når alle resultatene er skrevet ut kan man gå ut av TROLL (med `trexit`-kommandoen), og så skrive ut log-fila `troll.log` (denne fila er ei ren tekstfil) på papir. Denne log-fila vil være ganske stor, og vil inneholde en masse overflødige tegn, og blanke felter. det kan derfor være lønnsomt å hente inn log-fila i en editor (for eksempel `emacs`), og så utføre litt 'kosmetikk' på denne. Vekstprosentene vil uansett ligge i klartekst som kolonnevektorer, der vi har sektorvise tall loddrett. Vektorene med vekstprosenter kan så tas inn i annet analyseverktøy for videre behandling.

NB! Dersom observasjonen i en tidsserie er lik 0 i det året man skal regne ut vekstprosenter *fra*, kan man risikere at macroen vil dividere med 0. Vi har lagt inn en test for møte denne risikoen. Dersom den korresponderende observasjonen er lik 0 også for det neste året, er vekstprosenten lik 0, og den blir da satt lik denne verdien. Dersom observasjonen i det første året er lik null, men forskjellig fra null året etter, forsøker macroen å dividere med null. I så fall returnerer macroen verdien 'NA'.

DEL IV
Modellen ARBEID

1 Innledning

'Hovedmodellen' i REGARD kalles ARBEID. Når du skal utføre simuleringer må du først forvise deg om at modellen er korrekt, det vil si at koeffisientene som inngår i modellens likninger er riktige. Genereringen av likningene i modellen utføres ved å kalle på macroen `arbeid`, som så kaller på en rekke undermacroer. Macroen `nysysutd` genererer likninger for sysselsettingen etter kjønn, utdanning, alder og region. Macroen `gam2` genererer likninger som sikrer at det totale antall sysselsatte er likt, enten man splitter opp etter kjønn, alder, utdanning og region, eller over region og sektor. Macroen `ettersp2` lager likninger som bruker tall for netto pendling for å beregne markedsleie, mens macroen `nyutd` lager likninger som fremskriver befolkningen etter utdanning. Til dette hentes tall blant annet fra MOSART.

De tre macroene `flyttmod`, `flytmod2` og `flytmod5` lager likninger for flytting etter ulike personkjenntegn (kjønn, alder og utdanning). Macroen `innutv` genererer likningene for inn- og utvandring. Nye tall for antall syssesatte blir beregnet i likningene generert av macroen `nysyss`. Til slutt har vi macroen `nytilbud`. Denne brukes for å lage likninger for tilbud av arbeid, hvor variablene for arbeidsstyrke etter ulike personkjenntegn inngår.

2 Variable i ARBEID

Under følger en oversikt over alle variable, endogene og eksogene, samt alle koeffisienter som inngår i modellen ARBEID. Mange av variablene er indeksert over ulike personkjenntegn, som kjønn, alder, og utdanning. I tillegg, kan variable også være indeksert over sektor og region. Vi har innført følgende konvensjon for indeksering av de ulike kjennetegnene:

- `i` - Kjønn.
- `j` - Utdanning.
- `k` - Region.
- `l` - Alder.
- `m` - Sektor.

Dette systemet går igjen helt konsekvent både i denne dokumentasjonen, og i kildekode til macroene som genererer likningene i ARBEID. Dersom det er flere indekser for en type (gjelder spesielt region) i samme macro, er de ulike indeksene nummerert, for eksempel `k1`, `k2`, ... osv. Indeksen for kjønn er enten lik 1 eller 2 (menn eller kvinner). Indeksen for utdanning er enten lik 1, 2 eller 3, henholdsvis lav, midt og høy utdanning. Regionindeksen `k` løper fra 1 til 7 i henhold til REGARDs regioninndeling (se appendiks A). Indeksen for alder kan løpe fra 1 til 11 i henhold til REGARDs flerårige aldersinndeling. I mange situasjoner løper denne variabelen kun fra 3 til 9. Dette er de aldersgruppene som regnes som yrkesaktive (16-74 år). Sektorindeksen følger MODAGs sektorinndeling i 28 sektorer. For en oversikt over disse, se appendiks A.

I oversikten under gis det også informasjon om hva slags størrelser variablene representerer, hvilke personkjenntegn de er indeksert over, og hvilke macroer de inngår

i. For en enda mer detaljert oversikt over hvordan de ulike variablene inngår i likningene henvises til seksjon 4, hvor strukturen på likningene i ARBEID og deres tolkning diskuteres.

2.1 Endogene variable

- 1 - Sysselsatte etter kjønn, utdanning og (arbeidssteds-)region. Dette gir $2 \cdot 3 \cdot 8 = 48$ likninger. Disse er generert av macroen(e) `nysyssutd` (og `ettersp2`).
- `mgam` - Endogen skaleringsfaktor for beregning av sysselsetting (etter region) som sørger for at den totale sysselsettingen er lik uansett hvordan den splittes opp etter personkjennetegn, sektorer og regioner. Dette gir 8 likninger (inkludert ekstraregionen), generert av macroen `gam2`.
- `np` - Netto pendling etter kjønn, utdanning og region. Dette gir $2 \cdot 3 \cdot 7 = 42$ likninger, laget av macroen `ettersp2`.
- `jnp` - Skaleringsfaktor for beregning av pendling etter kjønn og utdanning. Gir $2 \cdot 3 = 6$ likninger generert av macroen `ettersp2`.
- `xm1` - Markedsleie etter kjønn, utdanning og region, noe som også gir 42 likninger. Disse likningene er også generert av `ettersp2`.
- `butg` - Folketall etter kjønn, alder, utdanning, og region. Oppdatering etter utdanning. Derfor kun aldersgruppe 3 – 9 dvs eldre enn 16 år. Dette gir $2 \cdot 3 \cdot 7 \cdot 7 = 294$ likninger. Lages av macroen `nyutd`.
- `jb` - Skaleringsfaktor for beregning av folketall, slik at det endogene beregnede folketall (BUTG) summert over utdanningskategorier stemmer overens med det eksogene folketall for hver kombinasjon av kjønn, alder og region. Dette gir $2 \cdot 7 \cdot 7 = 98$ likninger. Lages også i macroen `nyutd`.
- `ut` - Utflyttere etter kjønn, alder, utdanning, og 'fra'-region. Gir som i tilfellet over 294 likninger som lages av macroen `flyttmod`.
- `in` - Innflyttere etter kjønn, alder, utdanning, og 'til'-region. Totalt 294 likninger som genereres i macroene `flyttmod` og `flyttmod2`.
- `utb` - Utflytting av barn etter kjønn, alder og region. For aldersgruppe 1 og 2. Gir derfor $2 \cdot 2 \cdot 7 = 28$ likninger, generert i macroen `flyttmod5`.
- `inb` - Innflytting av barn etter kjønn, alder og region. På samme måte som over gir dette 28 likninger, også generert av macroen `flyttmod5`.
- `flrb` - Flytterater for barn etter kjønn, alder, fra- og tilflyttingsregion. Gir $2 \cdot 2 \cdot 7 \cdot 6 = 168$ likninger, laget av `flyttmod5`.
- `inv` - Innvandring etter kjønn, alder og region, totalt $2 \cdot 9 \cdot 7 = 126$ likninger, generert av `innutv`. Inngår også i oppdatering av befolkning i macroen `nyutd`.
- `utv` - Utvandring etter kjønn, alder og region. Også 126 likninger, laget av `innutv`. Brukes også i `nyutd`.
- `bruttoin` - Brutto innvandring. Likningen er laget av `innutv`.

- `syss` - Nye sysselsettingstall etter (bosteds-)region. Sju likninger som lages av macroen `nysyss`.
- `as` - Arbeidsstyrke etter kjønn, alder, utdanning og region. Dette gir $2 \cdot 7 \cdot 3 \cdot 7 = 294$ likninger, generert av macroen `nytilbud`.
- `jas` - Skaleringsfaktor ved beregning av arbeidsstyrke. Sørger for at den totale arbeidsstyrke summert over personkjennetegn er lik den totale arbeidsstyrken hentet fra MODAG. Denne likningen er generert av `nytilbud`.

2.2 Eksogene variable

- `gamma` - Korreksjonsvariabel (trendledd) etter kjønn, utdanning og region ved beregning av arbeidsstedets sysselsetting. Brukes i macroene `nysyssutd` og `ettersp2`.
- `ls` - Sysselsetting etter region og sektor. Beregnes i Formodellen på bakgrunn av sysselsettingstall hentet fra MODAG. Inngår i macroene `nysyssutd`, `gam2` og `ettersp2`.
- `s.np` - Korreksjonsfaktor pendling. Brukes i macroen `ettersp2`.
- `m` - Variabel som inneholder vekstrater for aldersgrupper. Hentes fra MOSART, og brukes i macroen `nyutd`.
- `b` - Eksogent befolkningstall, som brukes i flere macroer: `nyutd`, `flyttmod`, `flytmod2`, `flytmod5`, `innutv` og `nytilbud`.
- `s.utb` - Korreksjonsfaktor, utvandring barn.
- `s.inb` - Korreksjonsfaktor, innvandring barn.
- `s.ut` - Korreksjonsfaktor, utvandring.
- `s.in` - Korreksjonsfaktor, innvandring.
- `antot` - Total arbeidsstyrke beregnet i macroen `arbstyrk` i formodellen, basert på den totale arbeidsstyrken hentet fra MODAG (KVARTS), justert for vernepliktige og utlendinger i utenriks sjøfart. Brukes i macroen `nytilbud`.
- `s.as` - Korreksjonsfaktor arbeidsstyrke (`nytilbud`).

2.3 Koeffisienter

- `nyut` - Sysselsettingsandeler etter kjønn, utdanning, og region i hver sektor. Brukes i `nysyssutd`.
- `lpers` - Sysselsettingsandeler for ekstrarfylke (tilsv `nyut`), som brukes i `ettersp2`.
- `npnd` - Pendlingsandeler etter kjønn og utdanning. Brukes i `ettersp2`.
- `utind` - Indekser for vekst i utdanning. Brukes i macroen `nyutd`.
- `utford` - Fordeling av personer med høyere utdanning i alder 16–19 år. Brukes også i `nyutd`.

- `flyt` - Flytterater for alder 16 – 19 år. Brukes i macroen `flyttmod`.
- `flp` - Estimerte parametre i flyttemodellen. Brukes i macroene `flyttmod`, `flytmod2` og `flytmod5`.
- `bflin` - Flytterater barn. Brukes i `flytmod5`.
- `utvan` - Fordeling av utvandring på kjønn og alder. Brukes i `innutv`.
- `invan` - Fordeling av innvandring på kjønn og alder. Brukes i `innutv`.
- `ekstra` - Fordeling av sysselsatte fra ekstrasfylket (på andre regioner). Brukes i macroen `nysyss`.
- `ypind` - regionale forskjeller i yrkesprosenter. Brukes ved beregning av arbeidsstyrke i macroen `nytilbud`.

3 Generering/endring av modellen

Før vi skal kjøre REGARD må vi som nevnt forvise oss om at modellen er korrekt, dvs at koeffisientene som inngår er riktige. Endringer av koeffisientene skjer nokså sjelden. Det fins et opplegg i SAS for å lage nye koeffisienter, men dette er lite strømlinjeformet, og ganske krevende å bruke. Hver gang settet av koeffisienter endres må modellen genereres på ny. Dette gjøres ved at den gamle modellen først slettes. Deretter blir likningene generert på nytt, en operasjon som utføres med et kall på macroen `arbeid`. Mer formelt består generering av ny modell av følgende punkter.

- Slett eksisterende modell, med UNIX-kommandoen `rm arbeid.mod`
- Lag nye koeffisienter, og kopier disse til koeffisient-databasen `koeff.tr1`.
- Kjør macroen `arbeid` i TROLL, som oppretter modellen på nytt, og genererer nye likninger.

Det er viktig å slette den 'gamle' modellen først, for i motsatt fall vil den nye modellen slås sammen med den gamle. Macroen `arbeid` inneholder kommandoer for søk til databasen med (nye) koeffisienter. Oppgradering av modellen er ikke noe som skjer særlig ofte, men det er viktig å huske på å gjøre dette hver gang det lages nye koeffisienter.

4 Likningsstrukturen i ARBEID

I de følgende avsnittene skal vi gå igjennom macroene som genererer likningene i modellen ARBEID, og se på strukturen i de ulike likningene. Vi vil ikke studere TROLL-syntaksen i detalj her, men utskrifter av alle macroene er lagt ved dette notatet. For mer informasjon om strukturen på likningene, og tolkningen av disse, se Mohn et.al. [1994].

4.1 Litt om generering av likninger i TROLL

Generering av likninger i (nye) TROLL foregår ved hjelp av `modedit`-kommandoen, og underkommandoen `addeq`. Den generelle strukturen er som følger:

```
usemod regard;modedit;
addeq bottom
likn1,
likn2,
.
.
;
filemod;
```

Husk å få med to 'større-enn' tegn (>>) først på hver linje, dersom disse TROLL-setningene skal brukes i et program (macro). Merk at likningene adskilles med komma, og at `addeq`-kommandoen avsluttes med semikolon. Likningene angis nesten i 'klartekst' i henhold til TROLL's syntaks. For eksempel vil likningen

$$y = x^3 + 5z^2 - 24u + 17.88\sqrt{v}$$

i TROLL se ut som

```
y = x**3 + 5*(z**2) - 24*u + 17.88*sqrt(v),
```

Merk at multiplikasjon (`x*x`, `x*x*x` osv) er mer effektivt en potensering (`x**2`, `x**3` osv). Det er vanlig at flere likninger har samme struktur, der variablene som inngår er indeksert over en indeksemengde. I `REGARD` f.eks. er slike indeksemengder typisk *region*, *sektorkoder*, *utdanningskategorier*, *kjønn* og *alder*. Når man genererer likninger er det derfor vanlig at macroene har en løkkestruktur, hvor løkkeindeksen løper over den aktuelle indeksemengden, og hvor den aktuelle indeksen inkluderes i likningene på følgende måte:

```
for(i=1;i<=n;i=i+1)
begin;
  y&(i) = x&(i) + z&(i),
end;
```

Legg merke til at indeksene i likninger angis ved hjelp av `&`-notasjonen, akkurat som i gamle TROLL. For mer informasjon om macroprogrammering og generering av likninger henvises til TROLL-manualen.

4.2 Macroen `nysyssutd`

Macroen `nysyssutd` lager som nevnt likninger for sysselsettingen etter kjønn, utdanning og (arbeidssteds-)region. Strukturen på disse likningene er som følger:

$$l_{i,j,k} = mgam_k \cdot \gamma_{i,j,k} \cdot \sum_m ny_{i,j,k,m} \cdot ls_{m,k}$$

der summen er tatt over alle sektorene, $ls_{m,k}$ er sysselsettingen etter sektor og region, beregnet fra MODAG-tallene i formodellen. $\gamma_{i,j,k}$ er en eksogen variabel (trendledd) som angir hvordan sysselsettingen vil utvikle seg over tid som funksjon av kjønn, utdanning, og region. $mgam_k$ er en endogen variabel (skaleringfaktor), som brukes

for å justere sysselsettingen, slik at den blir lik enten man splitter opp etter sektor og region, eller etter kjønn, utdanning og region (se macroen `gam2`). $ny_{i,j,k,m}$ er en koeffisient bestemt på forhånd som angir den sektorvise andelen etter kjønn, utdanning og region. $l_{i,j,k}$ er den endogene sysselsettingen etter kjønn, utdanning og region. I nye TROLL vil genereringen av én likning se ut omtrent som følger:

```
>>l&(i)&(j)&(k)'n = mgam&(k)'n*gamma&(i)&(j)&(k)*(
for(m=1;m<=nr;m=m+1)
begin;
  se=values(sektor,m);
  ny = values(nyut,m);
  >>&(ny)*ls&(se)&(k)+
end;
>>0),
```

Variabelen `nr` er lik antall elementer i sektorlista. Likningen over sier altså at sysselsettingen etter kjønn, utdanning og region er lik et veiet snitt (med vekt $ny_{i,j,k,m}$) over sysselsettingen etter sektor og region, og skalert med faktoren $mgam_k$ (og $\gamma_{i,j,k}$) slik at den totale sysselsettingen ikke endres dersom den splittes opp på ulike måter.

4.3 Macroen `gam2`

Likningene generert av macroen `gam2` har struktur:

$$\sum_i \sum_j l_{i,j,k} = \sum_m l_{s_m,k}$$

Denne likningen sier oss at den totale sysselsettingen (over region) ikke skal endres uansett hvordan den splittes opp. I TROLL-syntaks ser dette ut som:

```
for(i=1;i<=2;i=i+1)
begin;
  for(j=1;j<=3;j=j+1)
  begin;
    >>l&(i)&(j)&(k)+
  end;
end;
>>0=
for(m=1;m<=nr;m=m+1)
begin;
  se = values(sektor,m);
  >>ls&(se)&(k)+
end;
>>0,
```

der `k` er indeksert over de åtte regionene (inkludert ekstraregionen), og `nr` er antall elementer i sektorlista, som i forrige macro.

4.4 Macroen `ettersp2`

Denne macroen lager likninger for å beregne etterspørsel etter arbeid. Første del lager likninger for sysselsettingen i ekstrarfylket, etter samme modell som i `nysyssutd`.

Forskjellen er koeffisienten. Istedenfor `nyut`, brukes her koeffisienten `lpers`. Likningene for å beregne netto pendling etter kjønn, utdanning og region har struktur:

$$np_{i,j,k} = (1 - npend_k) \cdot jnp_{i,j} \cdot l_{i,j,k} + s \cdot np_{i,j,k}$$

der $npend_k$ er en koeffisient ('pendlingsandelen'), $jnp_{i,j}$ er en justeringsvariabel som trengs for å sikre at den *totale* netto pendlingen blir lik null, og $l_{i,j,k}$ er sysselsettingen som beregnet i macroen `nysyssutd`. $s \cdot np_{i,j,k}$ er en korreksjonsfaktor. I TROLL-syntaks ser dette ut som

```
np=values(npend,k);
>>np&(i)&(j)&(k)'n = (1-&(np))*jnp&(i)&(j)'n*1&(i)&(j)&(k) +
>>s*np&(i)&(j)&(k),
```

Merk at justeringsvariabelen `jnp` settes lik 1 (ignorerer) dersom $np \leq 1$.

I `ettersp2` blir også likninger for markedsleie generert. Disse likningene har struktur

$$xml_{i,j,k} = \frac{(l_{i,j,k} - np_{i,j,k} + x \cdot l_{i,j,8})}{\sum_l as_{i,l,j,k}}$$

Her er $xml_{i,j,k}$ markedsleien etter kjønn, utdanning og region, $l_{i,j,k}$ sysselsetting (etter arbeidsstedsregion), $np_{i,j,k}$ netto pendling, $l_{i,j,8}$ sysselsettingen i ekstrasfylket, og $as_{i,l,j,k}$ er arbeidsstyrken splittet opp etter kjønn, alder, utdanning, og region. Koeffisienten x er variabelen som fordeler tallene for ekstrasfylket på de andre regionene. I TROLL-syntaks ser likningen for markedsleie ut som følger:

```
>>xml&(i)&(j)&(k)=(1&(i)&(j)&(k) - np&(i)&(j)&(k)
if (x <> 0) then
  >>+&(x)*1&(i)&(j)8
>>)/(
for(l=3;l<=9;l=l+1)
begin;
  >>as&(i)&(l)&(j)&(k)+
end;
>>0),
```

4.5 Macroen `nyutd`

Denne macroen lager likninger for befolkningen etter personkjennetegn. Likningene ser litt forskjellige ut, avhengig av utdanningskategori og alder. Det er jo slik at ikke alle aldersgruppene er representert i alle utdanningskategorier. For eksempel er det ytterst få yngre enn 20 år som har høyere utdanning. For slike unntak har man derfor spesialløsninger. For aldersgruppene 4 – 9 (20 – 74 år), har vi følgende struktur:

$$butg_{i,l,j,k} = ut \cdot m_{i,l,j} \cdot jb_{i,l,k} \cdot b_{i,l,j,k} - inv_{i,l,k} + utv_{i,l,k}$$

der $butg_{i,l,j,k}$ er folketall etter kjønn, alder, utdanning og region, ut er indekser for utdanningsvekst (`utind`), $jb_{i,l,k}$ er en justeringsvariabel for utdanningsfordelingen, $b_{i,l,j,k}$ er (eksogent) folketall som skal justeres (hentet fra demografidelen). $inv_{i,l,k}$ og $utv_{i,l,k}$ er henholdsvis innvandring og utvandring etter kjønn, alder og region.

For de i aldersgruppe 3 (16 – 19 år) med høyere utdanning (utdanningskategori 3) har vi følgende struktur:

$$butg_{i,3,3,k} = ut \cdot bm_{i,3,3}$$

Koeffisienten ut hentes fra vektoren *utford*, som sier hvor stor andel av aldersgruppe 3 som har høyere utdanning. Variabelen *bm* inneholder absolutt tilvekst i aldersgruppe 3. Macroen avsluttes med at vi summerer opp (aggregerer) alle variablene $butg_{i,l,j,k}$ over utdanningskategori, slik at vi får 'nye utgaver' av variablene *butg* som er splittet opp kun etter kjønn, alder og region.

4.6 Macroen flyttmod

Macroen *flyttmod* genererer likninger for flytting etter personkjenne tegn, dvs kjønn, alder, utdanning. Likningene har litt ulik struktur, avhengig av alder og utdanning, på samme måte som forrige macro. For de i aldersgruppe 3 (16–19 år), med høyeste utdanning har vi følgende struktur på likningene for ut- og innflytting:

$$ut_{i,3,3,k} = \left(\sum_{k_1=1}^7 \text{flytt}_{k_1} \right) \cdot \frac{b_{i,3,3,k}}{1000} + s.ut_{i,3,3,k}$$

$$in_{i,3,3,k} = \sum_{k_1=1}^7 \frac{\text{flytt}_{k_1} \cdot b_{i,3,3,k_1}}{1000} + s.in_{i,3,3,k}$$

der flytt_k er flyttekoeffisienter for en gitt region (for flytting til de andre), *b* er (ekso-gen) befolkning etter personkjenne tegn (kjønn, alder, utdanning og region), og *s.in* og *s.ut* er korreksjonsvariable. Utflyttingen fra region *k* er derfor essensielt lik summen av alle flyttekoeffisientene for flytting til de andre seks regionene multiplisert med befolkningen i den aktuelle persongruppen i region *k*. Innflyttingen til region *k* er lik flyttekoeffisienten for flytting fra en gitt region k_1 til region *k* multiplisert med befolkningen i aktuell gruppe i region k_1 , summert over alle regioner $k_1 \neq k$. For aldersgruppene 7–9 (45–74 år), er strukturen på likningene helt analog.

For alle i aldersgruppe 4–6 (20–44 år) og de i aldersgruppe 3 uten høyere utdanning har vi et noe annerledes og mer komplisert opplegg for utflytting. For å beregne utflyttingsandeler nyttes blant annet *markedsleien*, *xml*:

$$ut_{i,l,j,k} = b_{i,l,j,k} \cdot \frac{\sum_{k_1 \neq k} \exp(\text{flytt}_{k_1} \cdot \frac{\text{xml}_{i,j,k}}{\text{xml}_{i,j,k_2}})}{1 + \sum_{k_1 \neq k} \exp(\text{flytt}_{k_1} \cdot \frac{\text{xml}_{i,j,k}}{\text{xml}_{i,j,k_2}})} + s.ut_{i,l,j,k}$$

der k_1 og k_2 er hjelpeindekser som angir tilflyttingsregioner. Likningen for utflytting er altså på formen

$$ut_{i,l,j,k} = \frac{\psi(\text{xml}_{i,j,k})}{1 + \psi(\text{xml}_{i,j,k})} \cdot b_{i,l,j,k} + s.ut_{i,l,j,k}$$

der ψ er en funksjon av *markedsleien*. I TROLL-syntaks ser dette ut som:

```
>>ut&(i)&(l)&(j)&(k)=b&(i)&(l)&(j)&(k)*((
for(k1=1;k1<=6;k1=k1+1)
begin;
  fl = values(flytt,k1);
  if (k1<k) then
    if (fl<>0) then
      >>exp(&(fl)*xml&(i)&(j)&(k)/xml&(i)&(j)&(k1))+
```

```

else
  >>1+
else
begin;
  k2=k1+1;
  if (f1<>0) then
    >>exp(&(f1)*xml&(i)&(j)&(k)/xml&(i)&(j)&(k1))+
  else
    >>1+
  end;
end;
end;
>>0)/(
for(k1=1;k1<=6;k1=k1+1)
begin;
  fl = values(flytt,k1);
  if (k1<k) then
    if (f1<>0) then
      >>exp(&(f1)*xml&(i)&(j)&(k)/xml&(i)&(j)&(k1))+
    else
      >>1+
    else
begin;
  k2=k1+1;
  if (f1<>0) then
    >>exp(&(f1)*xml&(i)&(j)&(k)/xml&(i)&(j)&(k2))+
  else
    >>1+
  end;
end;
end;
>>1))+s.ut&(i)&(l)&(j)&(k),

```

if-testen `if (f1<>0) then` er lagt inn fordi i det tilfelle at $f1=0$ så vil det korresponderende eksponential-leddet reduseres til 1. Testen sørger derfor for at likningsstrukturen ikke blir unødige komplisert idet ledd på formen $\exp(0*\dots)$ settes lik sin numeriske verdi 1. Grunnen til at man må teste på at $k_1 \neq k$ er åpenbar, all den tid at flyttinger innenfor en region ikke blir registrert som flytting.

4.7 Macroen flyttmod2

Denne macroen lager likninger for innflytting til hver region, for de persongruppene som gjenstod fra macroen flyttmod, dvs aldersgruppe 4 – 6(20 – 44 år), og de i aldersgruppe 3(16 – 19 år) uten høyere utdanning. Strukturen på disse likningene er helt analoge til de tilsvarende likningene for utflytting for de samme persongruppene generert av flyttmod:

$$in_{i,l,j,k} = \sum_{k1 \neq k} \left\{ b_{i,l,j,k1} \cdot \frac{\exp(\text{flytt}_{k1} \cdot \frac{\text{xml}_{i,j,k1}}{\text{xml}_{i,j,k}})}{1 + \sum_{k2 \neq k1} \exp(\text{flytt}_{k2} \cdot \frac{\text{xml}_{i,j,k1}}{\text{xml}_{i,k,k2}})} \right\} + s.in_{i,l,j,k}$$

der vi i hvert ledd i summen over $k1 \neq k$ beregner andelen som flytter til region k , blant utflyttere fra region $k1$.

4.8 Macroen flytmod5

Macroen flytmod5 genererer likninger for flytting av barn, dvs aldersgruppe 1 og 2 (0–15 år). For utflytting er opplegget analogt med de to foregående macroene, men er delt i to, idet vi først beregner *flytteratene* flrb for barn fra region til region, for deretter å beregne utflyttingen som et veiet snitt av barnetallene multiplisert med flytteratene. Likningene for beregning av flytterater har følgende (relativt kompliserte) struktur:

$$\text{flrb}_{i,l,k,k1} = \text{bflin}_{i,l,k,k1} \cdot \frac{\sum_{j=1}^3 \left\{ b_{2,l,j,k} \cdot \frac{\exp(\text{flp}_{2,l,j,k,k1} \cdot \frac{\text{xm1}_{2,j,k}}{\text{xm1}_{2,j,k1}})}{1 + \sum_{k2 \neq k} \exp(\text{flp}_{2,l,j,k,k2} \cdot \frac{\text{xm1}_{2,j,k}}{\text{xm1}_{2,j,k2}})} \right\}}{\sum_{j=1}^3 b_{2,l,j,k}}$$

der $\text{bflin}_{i,l,k,k1}$ er en koeffisient som er med å bestemme fordelingen av flytteratene. Legg merke til at vi bruker befolkningstall for kvinner (kjønn = 2): $b_{2,l,j,k}$. Dette skyldes at tallene for barn avhenger av mødregenerasjonen, dvs flytteratene for barn bestemmes av flytteratene hos kvinner. Dermed har likningene for beregning av utflytting av barn følgende enkle form:

$$\text{utb}_{i,l,k} = \text{barn}_{i,l,k} \cdot \left(\sum_{k1 \neq k} \text{flrb}_{i,l,k,k1} \right) + \text{s.utb}_{i,l,k}$$

For innflytting blir likningene helt tilsvarende:

$$\text{inb}_{i,l,k} = \sum_{k1 \neq k} \{ \text{barn}_{i,l,k1} \cdot \text{flrb}_{i,l,k1,k} \} + \text{s.inb}_{i,l,k}$$

4.9 Macroen innutv

I denne macroen lages likningene for inn- og utvandring. Likningene for utvandring (av voksne) etter kjønn, alder og region:

$$\text{utv}_{i,l,k} = \text{utvan}_{i,l} \cdot \sum_{j=1}^3 \frac{b_{i,l,j,k}}{1000}$$

der $\text{utvan}_{i,l}$ er en koeffisient, $b_{i,l,j,k}$ er befolkning. For utvandring av barn har vi følgende struktur:

$$\text{utv}_{i,l,k} = \text{utvan}_{i,l} \cdot \frac{\text{barn}_{i,l,k}}{1000}$$

der $\text{barn}_{i,l,k}$ er barnetall. Her løper indeksen for alder over aldersgruppene 1 og 2. For brutto innvandring får vi dermed følgende likning:

$$\text{bruttoin} = \sum_{i,l,k} \text{utv}_{i,l,k} + \text{nettoin}$$

Tilsvarende som likningene over vil likningene for innvandring etter kjønn, alder, region se ut som:

$$\text{inv}_{i,l,k} = \text{invan}_{i,l} \cdot \frac{\text{bruttoin}}{100}$$

der $\text{invan}_{i,l}$ er en koeffisient.

4.10 Macroen nysyss

Her lages nye likninger for sysselsatte etter bosted:

$$\text{syss}_k = \sum_{i,j} \{l_{i,j,k} - np_{i,j,k} + x_{i,j} \cdot l_{i,j,8}\}$$

der syss_k er sysselsettingen etter (bosteds-)region, $l_{i,j,k}$ er sysselsettingen etter kjønn, utdanning og (arbeidssteds-)region, som beregnet i `nysyssutd`, np er netto pendling. Legg merke til at vi tar hensyn til ekstrasfylket, og fordeler sysselsatte herfra på de andre regionene, ved hjelp av koeffisienten `ekstra` ($x_{i,j}$).

4.11 Macroen nytilbud

I denne avsluttende macroen lages likninger for å beregne arbeidsstyrke. Likningen ser ut som følger:

$$as_{i,l,j,k} = \min\{b_{i,l,j,k} \cdot ypind_{i,l,j,k} \cdot yp_{i,l,j} + s.as_{i,l,j,k}, b_{i,l,j,k}\} \cdot jas$$

der as er arbeidsstyrken, $ypind$ er en koeffisient, b er en (eksogen) variabel med befolkningstall, $s.as$ er en korreksjonsvariabel, og jas er en skaleringsvariabel som skal sikre at den totale arbeidsstyrken, as , summert over alle personkjennetegn, er lik den totale arbeidsstyrken hentet fra MODAG. Dette er uttrykt ved følgende likning:

$$\text{antot} = \sum_{i,l,j,k} as_{i,l,j,k}$$

der vi summerer over kjønn, alder, utdanning og region.

DEL V

Demografiske oppdateringer

1 Innledning

Demografien i REGARD bygger på den såkalte *Leslie-modellen* (se Mohn et.al. [1994]) for populasjoner. Demografiske oppdateringer er helt essensielle i REGARD, da slike endringer beregnet av ARBEID må registreres, før de leses inn som eksogene størrelser til simuleringen neste år. Dette opplegget er nokså komplisert, fordi aldersgruppene her er *ettårige*, mens de i modellen ARBEID er flerårige, dvs befolkningen er delt opp i 11 aldersgrupper. I denne delen skal vi se nærmere på hvordan denne demografiske oppdateringen finner sted.

1.1 Hjelpemacroene colsplit og kombiner

Kommandoen `colsplit` i gamle TROLL eksisterer ikke lenger. Der hvor denne funksjonen ble brukt i det gamle systemet har vi vært nødt til å lage et nytt opplegg.

Setningen `DO COLSPLIT(X)`; i gamle TROLL splittet matrisen `X` opp i kolonnevektorer, hvor hver kolonne fikk navn `X.C&(i)` der `&(i)` er kolonnennummeret. Dette opplegget er nå erstattet av en løkkestruktur på formen:

```
>>do x=navn;
for(i=1;i<=nkol;i=i+1)
begin;
  >>do xc&(i)=submat(x,,col(x)==&(i));
end;
```

NB! Merk at understrekingen i navnet er fjernet. I nye TROLL betyr `x.c` som kjent at man skal lage en vektor (matrise) `c` på underkatalogen `x`. Vi har også laget en egen macro `colsplit`, som inneholder løkkestrukturen over. Syntaksen for å bruke denne er:

```
&colsplit; >>navn
```

Kommandoen `partcomb` har i nye TROLL syntaksen:

```
do y=partcomb(m,n,a11,...,a1n,a21,...,a2n,...,am1,...,amn);
```

Denne kommandoen lager en ny matrise ved å slå sammen matrisene a_{11}, \dots, a_{mn} , der n matriser blir slått sammen i 'bredden', og m i 'høyden'. Det er viktig at dimensjonen på matrisene er slik at en sånn sammenslåing er mulig, dvs at dimensjonene stemte overens: $a_{i,j}$ må ha like mange rader som $a_{i,j+1}$, og like mange kolonner som $a_{i+1,j}$. I nye TROLL eksisterer `partcomb` fra og med versjon 1.0. For å klare å utføre disse operasjonene i utviklingsperioden av REGARD-systemet, mens vi kjørte på tidligere versjoner enn 1.0 ble vi nødt til å lage en egen macro `kombiner` som gjorde jobben. I noen tilfeller har vi også kunnet erstatte bruken av denne macroen med en direkte bruk av `combine`-kommandoen. Dette avhenger først og fremst av størrelsen på parametrene m og n , samt dimensjonen på delmatrisene som inngår.

Dersom $n = 1$, og delmatrisene a_i er skalare størrelser betyr

```
DO Y=PARTCOMB(m,1,a1,...,am);
```

at vi skal lage en $m \times 1$ kolonnevektor der elementene er (de skalare størrelsene) a_i . Den gamle kommandoen kan derfor erstattes med:

```
>>do y=combine(
for(i=1;i<=m;i=i+1)
```



```
begin;  
  if (i<m) then  
    >>ai,  
  else  
    >>ai  
end;  
>>);
```

Dersom $m = 1$ og $n > 1$ (og hver a_i er en skalar) betyr det at vi ønsker å lage en radvektor. Dette kan gjøres helt analogt med det forrige tilfellet, bare at man transponerer vektoren produsert av `combine`-kommandoen.

Dersom matrisene $a_{i,j}$ ikke er skalare, men derimot matriser kommer vi ikke utenom å bruke `partcomb`-kommandoen, eller macroen `kombiner` i tidligere versjoner enn 1.0. Syntaksen for å bruke denne macroen er som følger:

```
&kombiner; >>navn x y a11a12...amn
```

For den som er interessert er en utskrift av macroene `colsplit` og `kombiner` lagt ved i dette notatet. Men husk at fra og med versjon 1.0 vil vi anbefale å bruke den innebygde `partcomb`-funksjonen.

2 Variable i REGARD's demografidel

Dataene i REGARDs demografidel består dels av demografitall som hentes utenfor REGARD, og dels tall for befolkning, innvandring og flytting beregnet i modellen ARBEID, og som trengs for å oppdatere de demografiske størrelsene. Noen av variablene blir brukt som eksogene variable i neste simulering i modellen ARBEID. Det blir også brukt endel hjelpevariable under oppsplitting fra REGARDs flerårige aldersgruppeinndeling, til den 'vanlige' ettårige inndelingen, og til aggregeringen den motsatte veien. I tillegg forekommer en rekke hjelpevariable i de ulike macroene. Disse brukes til å lagre mellomregninger.

I macroene som inngår, har vi brukt samme kodesystem som tidligere for å kjenne igjen ulike personkjennetegn på indeksvariable: `i` er kjønn, `j` er utdanning, `k` er region, og `l` er alder. For øvrig inneholder variabelen `fra` første simuleringsår, til siste simuleringsår, `aar0` aktuelt simuleringsår, og `aar1` forrige simuleringsår, dvs `aar0-1`. Alle disse variablene er på numerisk format. I noen situasjoner (i forbindelse med tidsserier), blir årstallene konvertert til tekstformat, fordi vi da må skrive f.eks. `1990A` istedenfor bare `1990`. Under følger en oversikt over de ulike variable som inngår i demografidelen. Vi har skilt mellom inn- og ut-variable til/fra macroene, variable for å lagre mellomregninger, og rene hjelpevariable (matriser) som inneholder indeksmengder, diverse rater, og fordelingstall.

2.1 Inn- og ut-variable

- `butg&(i)&(l)&(j)&(k)` - Endogen variabel fra modellen ARBEID, som inneholder befolkning etter kjønn, REGARDs flerårige aldersinndeling, utdanning og region. Brukes i macroen `ninv`.
- `bef&(j)&(aar)` - Ettårig befolkning etter utdanning ved inngangen til året. Henter inn verdiene for forrige år i macroen `ninv` for å beregne fordelingen når

- de flerårige tallene i foregående variabel skal splittes opp til ettårig struktur. For aktuelt simuleringsår inneholder variabelen den ettårige befolkningen etter utdanning inkludert innvandring og flytting *ved inngangen til året*, og er en ut-variabel i macroen `flytt`.
- `inv&(i)&(1)&(k)`, `utv&(i)&(1)&(k)` - Henholdsvis innvandring og utvandring etter kjønn, alder og region, endogent beregnet i modellen ARBEID. Slås sammen til vektorer og matriser i macroen `ninv`.
 - `bhat&(aar)` - rendemografisk fremskrivning av ettårig befolkning ved utgangen av året. Beregnet i forrige runde (output fra macroen `rb`), og brukes som input i `ninv`.
 - `nhat&(aar)` - 172×7 matrise som inneholder korrigerede regionale ettårige befolkningstall, inkludert netto innvandring, dvs `nhat&(aar) = bhat&(aar-1) + ninv&(aar)`. Beregnes i `ninv`, og brukes som inn-variabel i macroen `flytt`.
 - `nhat&(j)` - Tilsvarende som foregående variabel, men bare med tallene for yrkesaktive aldersgrupper, dvs 16 – 74 år, og etter utdanningskategori. Er derfor en 118×7 matrise. Blir oppdatert med variabelen `ninvu&(j)`. Beregnes også i `ninv`, og er input til `flytt`.
 - `in&(i)&(1)&(k)` - Endogent beregnede tall for innflytting fra ARBEID. Slås sammen til vektorer i `flytt`.
 - `ut&(i)&(1)&(k)` - Endogent beregnede tall for utflytting fra ARBEID. Slås sammen til vektorer i `flytt`.
 - `inb&(i)&(1)&(k)` - Endogent beregnede tall for innflytting av barn fra ARBEID. Brukes også i `flytt`.
 - `utb&(i)&(1)&(k)` - Endogent beregnede tall for utflytting av barn fra ARBEID. Brukes i `flytt`.
 - `nfb&(aar)` - Endelige ettårige netto flyttetall, slått sammen til 172×7 matriser. Output fra macroen `flytt`.
 - `nfb&(j)&(aar)` - Ettårig flyttetall etter utdanning. 118×7 matrise som blir beregnet i `flytt`.
 - `bef&(aar)` - 172×7 matrise som inneholder befolkningen oppdatert med netto flytting i macroen `flytt`. Brukes som input i macroen `newborn`.
 - `bb&(aar)` - Variabel hvor de ettårige befolkningstallene fra `flytt` hentes inn i `newborn`. Ut av `newborn` inneholder denne variabelen den ferdig aldrede ettårige befolkningen. Input til både `rb` og `innbef`.
 - `befut&(j)&(aar)` - Ettårig befolkning etter utdanning, aldrede i `newborn`. Input til `rb` og `innbef`.
 - `b&(i)&(1)&(k)` - Befolkningstallene etter kjønn, alder og region konvertert til en tidsserie med årlig periodisitet og en observasjon. Lages i `innbef`.

- $b(i)(l)(j)(k)$ - Tilsvarende som $b(i)(l)(k)$, men fordelt etter utdanning. Lages også i `innbef`, og sammen med foregående variabel brukes denne som eksogen variabel i neste runde av ARBEID.
- $bm(i)33(k)$ - Tilsvarende som forrige variabel, men for de i aldersgruppe 3 (16–19 år) med høyere utdanning. Lages også i `innbef`, og brukes i ARBEID.
- $barn(i)(l)(k)$ - Tallene for barn. Lages i `innbef` og brukes som eksogen variabel i macroen `flytmod5` i ARBEID.
- $bh(aar)$ - Lokal variabel som inneholder befolkningstall i macroen `rb`.
- $butg(i)(l)(k)$ - Befolkning etter kjønn, alder og region, brukes som eksogen variabel (input) i neste runde av ARBEID. Blir beregnet i `utbef` ved hjelp av den rendemografiske fremskrivningen $bhat(aar)$.
- $bhat(aar)$ - Rendemografisk befolkningsfremskrivning etter ettårig alder med alderskjennetegn ved utgangen av året, eller ved inngangen til neste år om man vil. Beregnes i `rb`, der dødeligheten blir inkludert i ut-variabelen $bb(aar)$ beregnet i `newborn`. Brukes i flere macroer, men er først og fremst input til demografidelen i neste simuleringsrunde, dvs til macroen `ninv`.

2.2 Mellomregninger

- $khat(j)(k)$ - Vektorer som inneholder tall fra input-variabelen `butg` i macroen `ninv`, slått sammen etter kjønn og alder som 14×1 vektorer (7 aldersgrupper og 2 kjønn). Det er en vektor for hver utdanningskategori j og hver region k , dvs $3 \cdot 7 = 21$ vektorer. Denne variabelen brukes i macroen for netto innvandring, `ninv`.
- $fler(j)(k)$ - Ettårig befolkning etter utdanning, aggregert til flerårige grupper. Brukes i `ninv` for å beregne ny fordelingsmatrise mellom ettårige og flerårige grupper.
- `fler` - Foregående variabel slått sammen over utdanning og region til 118×14 matrise.
- $nhu(j)(k)$ - Ettårig eksogen befolkning etter utdanning 'fordelt' til den tilhørende flerårige gruppen. Ordnet som 188×14 matriser. Brukes i `ninv` til å beregne ny versjon av fordelingsmatrisen $ggu(j)(k)$ mellom ettårig og flerårige aldersstruktur.
- $bhat(j)(k)$ - Ny midlertidig befolkning etter ettårig alder beregnet ved å splitte opp matrisen(e) $khat(j)(k)$ ved hjelp av fordelingsmatrisen $ggu(j)(k)$. Dette gir en 118×1 vektor for hver utdanningskategori og region. Beregnes i macroen `ninv`.
- $bhat(j)$ - Foregående variabel slått sammen over region (118×7 matrise). Korrigert med utdanningsfordelingen gitt av $bef(j)(aar)$, deretter aggregert til variabelen `buhat`, og til slutt beregnet slik at den er konsistent med den totale eksogene befolkningen (rendemografisk fremskrivning $bhat(aar1)$) ved utgangen av foregående år. Blir også beregnet i `ninv`.

- $invx$ - 22×7 matrise med variablene for innvandring fra modellen ARBEID, slått sammen etter alder og region. Beregnes i $ninv$.
- utv - Tilsvarende som forrige variabel, men tall for utvandring.
- $ninv$ - netto innvandring etter alder og region. Først beregnes en 22×7 versjon, som differansen mellom de to foregående variablene. Deretter lages en ettårig versjon ved hjelp av oppsplittingsmatrisen $gg(k)$. Denne variabelen, $ninv(k)$ består av sju 172×1 vektorer, en for hver region.
- $khat(k)$ - 22×1 vektor som inneholder aggregerte befolkningstall for hver region over flerårig aldersinndeling (11 aldersgrupper og 2 kjønn). Beregnes også i $ninv$.
- $ghat(k) = 1/khat(k)$. 22×1 vektor som gir befolkningsandelen i hver av REGARDS aldersgrupper. Brukes i $ninv$ til å beregne fordelingsmatrisen $g(k)$.
- $bg(k)$ - 172×22 matriser bestående av 22 kopier av kolonne nr. k i $bhat(aar)$, elementvis multiplisert med hjelpematriksen cc . Gir fordeling av netto innvandring over ettårig alder, da vi antar at denne samsvarer med aldersstrukturen i befolkningen. Brukes i macroen $ninv$.
- $ninv(aar)$ - 172×7 matrise som består av de regionale kolonnene $ninv(k)$ slått sammen. Lages i macroen $ninv$.
- $ninvu(j)$ - 118×7 matrise som inneholder tall for netto innvandring etter utdanning. Inneholder kun tall for de aldersgrupper som er yrkesaktive, dvs 16 – 74 år. Lages også i macroen $ninv$.
- $infe(j)(k)$ - Innflytting etter utdanning j og region k . 14×1 vektor over kjønn og utdanning bygget opp av tallene for innflytting endogent beregnet i ARBEID. Beregnes i macroen $flytt$.
- $utf(j)(k)$ - Utflytting etter utdanning j og region k . 14×1 vektor over kjønn og utdanning bygget opp av tallene for utflytting endogent beregnet i ARBEID.
- $nfke(j)(k)$ - Netto flytting, differansen av de to foregående variablene.
- $nfb(j)(k)$ - 118×1 vektor som inneholder netto flytting etter ettårig alder for alder 16 – 74 år, etter utdanning j og region k . Korrigeres slik at total netto flytting for gitt kjønn, alder og utdanning er lik 0.
- $nf1$ - Variabel som brukes til å korrigere flyttetallene slik at total netto flytting blir lik 0.
- $nfbv$ - Voksnes flyttinger. Matrisene $nfb(j)$ aggregert over utdanning.
- $nfkb(i)(k)$ - Netto flytting barn, etter flerårig alder (aldersgruppe 1 – 2).
- $nfb(i)(k)$ - 16×1 vektor som inneholder tall for netto flytting av barn, oppsplittet etter ettårig alder (0 – 15 år).

- $\text{flyttb}(i)$ og $\text{fb}(i)$ - Variable for å korrigere barnas flyttinger slik at total netto flytting av barn også er lik null. Brukes i `flytt`.
- nfbg - Eldres flyttinger ($\equiv 0$). Brukes i `flytt`.
- $\text{nffb}(k)(\text{aar})$ - 172×1 vektor som inneholder tall for netto flytting etter ettårig alder. For flytting av eldre har vi lagt inn nuller i de aktuelle årsklassene (variabelen nfbg). Beregnes i `flytt`.
- fm - Antall nyfødte gutter etter region. 1×7 radvektor. Beregnes i `newborn`.
- fk - Antall nyfødte jenter etter region. 1×7 radvektor. Beregnes i `newborn`.
- ga1 og ga2 - Midlertidig fremskrevne eldre, henholdsvis menn og kvinner. Brukes i `newborn` når befolkningen aldres.
- b1 og b2 - Midlertidige befolkningsvariable som brukes når befolkningen aldres i `newborn`.
- $\text{k}(k)$ - 172×7 befolkningsmatrisen $\text{bb}(\text{aar})$ beregnet i `newborn` splittet opp i kolonner, og aggregert til REGARDs flerårige aldersgruppeinndeling (ved hjelp av matrisen cc), til 22×1 vektorer. Beregnes av macroen `innbef`.
- $\text{k}(\text{aar})$ - Foregående variabel slått sammen over region, til en 22×7 matrise. Beregnes i `innbef`.
- $\text{k}(i)(\text{aar})$ - Foregående variabel splittet opp etter kjønn i to 11×7 matriser.
- $\text{bgm}(1)(i)(k)$ - Befolkningstallene for eldre (aldersgruppe 10–11) ekstrahert fra forrige variabel.
- $\text{ku}(i)(j)(k)$ - Befolkningstall fordelt etter utdanning. Beregnes i `innbef`.
- $\text{bh}(\text{aar})$ - Lokal variabel som brukes i `rb` til å hente inn variabelen $\text{bb}(\text{aar})$ (fra `newborn`), og så inkludere dødeligheten i denne.
- $\text{kh}(k)(\text{aar})$ - Inneholder rendemografiske oppdaterte befolkningstall ($\text{bhat}(\text{aar})$) aggregert tilbake til REGARDs flerårige aldersgruppeinndeling. Beregnes i `utbef`.
- $\text{kh}(\text{aar})$ - Foregående variabel slått sammen over region til en matrise.

2.3 Hjelpvariable

Hjelpematriser som brukes ved aggregering fra ettårig til flerårig aldersfordeling, eller oppsplitting den andre veien.

- cc - 172×22 matrise bestående av 1'ere og nuller, og som beskriver overgangen mellom REGARD's flerårige aldersgrupper, og den ettårige aldersstrukturen i demografidelen. For alle 11 aldersgrupper i REGARD, og alle 86 ettårige aldersgrupper.

- cu - 118×14 matrise bestående av 1'ere og nuller, og som beskriver overgangen mellom REGARD's flerårige aldersgrupper, og den ettårige aldersstrukturen i demografidelen. Kun for de sju aldersgruppene med yrkesaktive (aldersgruppe 3 – 9), som tilsvarer alder 16 – 74 år i ettårig inndeling.
- $cbarn$ - 16×2 matrise bestående av 1'ere og nuller, og som beskriver overgangen mellom ettårig og flerårig alder for aldersgruppe 1 – 2 (0 – 15 år).
- $ggu(j)(k)$ - 118×14 matrise som gir fordelingen fra flerårig til ettårig aldersinndeling for befolkning etter utdanning. Lages bl.a. ved hjelp av matrisen $bef(j)(aar)$ (ettårig befolkning etter utdanning) og hjelpematriksen cu i macroen $ninv$.
- $g(k)$ - 172×1 vektorer som er lik matriseproduktet av $bg(k)$ og $ghat(k)$, og som inneholder nye fordelingstall for å beregne netto innvandring oppsplittet etter ettårig alder. Antar at fordelingen for netto innvandring er lik som aldersfordelingen i befolkningen. Beregnes i $ninv$.
- $gg(k)$ - 172×22 matriser som inneholder 22 kopier av vektoren $g(k)$, elementvis multiplisert med hjelpematriksen cc . Fordelingsmatrisen som brukes for å splitte opp netto innvandring etter flerårig alder til ettårig struktur.
- $fv(j)$ - Matrise for oppsplitting av flytting etter flerårig alder til ettårig struktur. Etter utdanning.
- $flb(i)$ - 16×1 vektor som inneholder fordelingstall mellom ettårig og flerårig aldersgruppeinndeling, for flytting av barn.
- $ffb(i)$ - 16×2 matrise lik det elementvise produkt av fordelingsvektoren $flb(i)$ og hjelpematriksen $cbarn$. Fordelingsmatrisen som angir overgangen mellom ettårig og flerårige flyttetall for barn. Brukes i $flytt$.

Andre hjelpevariable, som brukes til å plukke ut delmatriser som brukes i forskjellige sammenhenger.

- amk - Vektor som inneholder indeksene til de yrkesaktive i ettårig aldersfordeling (16 – 74 år).
- $menn$ - Inneholder indekser for menn.
- $kvinner$ - Inneholder indeksene som gir tallene for kvinner.
- bk - Vektor med indekser for de kvinner som føder barn, dvs 15 – 44.
- kk, km - Indekser for henholdsvis menn og kvinner i vektorer og matriser definert over REGARD's (flerårige) aldersgruppeinndeling.
- kku, kmu - Tilsvarende som forrige, men kun for de yrkesaktive.
- $ol(aar)$ - 172×1 vektor med overlevelsesserater for menn og kvinner. Er en sammenslåing av de to eksogene variablene sm og sk .
- fbm - 86×7 matrise som inneholder regionale fødselsrater for gutter. Siden kvinner i alderen 15 – 44 år antas å være de eneste som føder barn, er alle andre rater lik 0.

- fbk - 86×7 matrise som inneholder regionale fødselsrater for jenter. Siden kvinner i alderen 15 – 44 år antas å være de eneste som føder barn, er alle andre rater lik 0.
- sk, sm - 86×1 vektorer som inneholder overlevelsesserater i ulike aldergrupper for henholdsvis kvinner og menn.

3 Macroene i demografidelen

Gangen i REGARDs demografidel er omtrent som følger:

1. Endogen flerårig befolkning etter utdanning ved inngangen til året (eksl. innvandring/flytting), $butg(i)(1)(j)(k)$, beregnet ved simulering av modellen ARBEID for året før, hentes inn, og settes sammen på matriseform i variabelen $khat(j)(k)$. Denne befolkningen splittes først opp til ettårig alder ved hjelp av ettårig befolkning etter utdanning beregnet året før ($bef(j)(aar)$). Vi beregner deretter den endogene (ettårige) befolkningen etter utdanning, slik at den blir konsistent med total (ettårig) rendemografisk fremskrivning av befolkningen $bhat(aar)$ ved inngangen til året.
2. Endogene variable $inv(i)(1)(k)$ og $utv(i)(1)(k)$ for innvandring og utvandring hentes fra ARBEID, og befolkningstallene oppdateres med disse. Disse to første punktene utføres av macroen *ninv*.
3. Endogene variable for flytting, $in(i)(1)(j)(k)$, $ut(i)(1)(j)(k)$, $inb(i)(1)(k)$, og $utb(i)(1)(k)$ hentes inn, og oppdaterer befolkningen ved inngangen av året med disse. Lager derfor nye utgaver av variablene $bef(aar)$ og $bef(j)(aar)$. NB! Har lagt inn en konsistenssjekk som sikrer at netto flytting for gitt kjønn, alder og utdanning er lik 0. Dette punktet utføres av macroen *flytt*.
4. Dersom første simuleringsår er 1996 eller tidligere, erstattes fremskrevne befolkningstall med historiske for dette første simuleringsåret. Dette gjøres i macroen *dgraf*.
5. Den befolkningen vi nå har er befolkningen ved inngangen av året inkludert innvandring og flytting, og kjennetegnet med alder ved inngangen til året. Det neste som gjøres er å aldre befolkningen, dvs gi de alderskjennetegn som ved utgangen av året, samt inkludere en kohort med nyfødte. Macroen *newborn* gjør dette.
6. Vi bruker så befolkningen beregnet i foregående punkt til å beregne de eksogene variablene $b(i)(1)(j)(k)$, $b(i)(1)(k)$ og $barn(i)(1)(k)$ til neste kjøring av modellen. Dette gjøres av macroen *innbef*.
7. Deretter oppdateres befolkningen med dødelighet (macroen *rb*), og den befolkningen vi da står igjen med, rendemografisk ettårig fremskrivning av befolkningen $bhat(aar)$, brukes delvis i macroen *utbef* til å lage en eksogen variabel $butg(i)(1)(k)$ til ARBEID, og delvis som input i neste runde av demografidelen.

De foregående punkter kjøres ved å kalle på macroen `dgraf`, som igjen kaller på flere undermacroer som utfører selve oppdateringen av de ulike demografiske størrelsene. Vi skal her se nærmere på hvordan de ulike (under)macroene fungerer. Se også utskriften av macroene, som er påført kommentarer som beskriver funksjonsmåte.

3.1 Macroen `ninv`

Macroen `ninv` henter den rendemografiske fremskrivningen fra utgangen av foregående år, `bhat&(aar-1)`, for deretter å korrigere denne for netto innvandring i løpet av foregående år. Gangen i denne macroen er omtrent som følger:

1. Først innhentes de endogent beregnede befolkningstallene etter REGARDs flerårige aldersstruktur, `butg&(i)&(1)&(j)&(k)` fra ARBEID. Disse slås sammen til 14×1 -vektorer `khat&(j)&(k)` (7 aldersgrupper etter utdanning, 2 kjønn).
2. Deretter lages 118×1 vektorene `bhat&(j)&(k)` som er de flerårige endogene befolkningstallene oppsplittet til ettårig aldersfordeling. Denne oppsplittingen skjer ved hjelp av befolkning etter utdanning ved utgangen av foregående år, `bef&(j)&(aar-1)`.
3. Den ettårige befolkningen gjøres konsistent med den (ettårige) rendemografiske fremskrivningen av befolkningen ved utgangen av foregående år, `bhat&(aar-1)`.
4. De endogene variablene `inv&(i)&(1)&(k)` og `utv&(i)&(1)&(k)` (innvandring og utvandring) fra ARBEID settes så sammen til matriser, og netto innvandring over REGARDs aldersgrupperinndeling beregnes som en differanse av disse matrisene. Netto innvandring blir så splittet opp til ettårig alder på samme måte som befolkningstallene.
5. Til slutt blir både den totale befolkningen, og befolkningen etter utdanning korrigert for netto innvandring. Vi antar at det er samme utdanningsfordeling på nettoinnvandringen som i befolkningen.

Sammenslåingen av de endogent beregnede befolkningstallene `butg` gjøres i en for-løkke over fire nivåer; *kjønn*, *alder*, *utdanning* og *region*, og systematisk bygger opp variablene (vektorene) `khat&(j)&(k)` ved hjelp av `combine`-kommandoen i TROLL. Som nevnt splittes variablene `khat&(j)&(k)` opp etter ettårig aldersstruktur, ved hjelp av fordelingen over aldersgrupper i variabelen `bef&(j)&(aar1)`, ettårig befolkning etter utdanning ved inngangen til foregående år. Kort sagt gjøres dette ved å dividere befolkningen i en ettårig kohort på befolkningen i den flerårige aldersgruppen kohorten tilhører. Dermed får vi kohortens andel av hele aldersgruppen. Når vi så multipliserer befolkningen i en flerårig aldersgruppe fra ARBEID med andelen til en gitt kohort (tilhørende denne gruppen), får vi den ettårige befolkningen til den gitte kohorten. Andelene i selve oppsplittingen blir lagret på 118×14 matrisene `gg&(j)&(k)`, og den ettårige befolkningen blir midlertidig lagret i 21 stk. 118×1 vektorer `bhat&(j)&(k)`, der $j=1, 2, 3$ og $k=1, \dots, 7$, slått sammen over region til 3 stk. 118×7 matriser `bhat&(j)`.

Teknisk sett foregår oppsplittingen beskrevet over i en dobbelt for-løkke; over utdanning og region. Siden vi hele tiden betrakter befolkningen etter utdanning og region, må vi først splitte 118×7 matrisen `bef&(j)&(aar)` opp i 118×1 vektorer. De

ettårige tallene (etter utdanning og region) blir aggregert til flerårig aldersstruktur ved å multiplisere med matrisen cu (se beskrivelse av denne lenger opp). Resultatet lagres i variabelen $fler\&(j)\&(k)$:

```
>>dofile fler\&(j)\&(k) = matmult(transp(cu),bef\&(j)\&(aar1)c\&(k));
```

der $bef\&(j)\&(aar1)c\&(k)$ betyr kolonne nr. k i matrisen med ettårige befolkningstall etter utdanning. Variablene $fler\&(j)\&(k)$ er altså 14×1 vektorer (7 aldersgrupper og 2 kjønn). Hver slik vektor blir så omgjort til 118×14 matriser, ved at hver rad er lik den transponerte av den gitte vektoren. Kommandoen `>>do y = matmult(x1,ones)`;

der $x1$ inneholder kolonne nr. k i matrisen med ettårig befolkning etter utdanning, og $ones$ er en 1×14 radvektor bestående utelukkende av 1'ere, lager derfor en 118×14 matrise med den ettårige befolkningen etter utdanning i hver kolonne. Kommandoen

```
>>dofile nhu\&(j)\&(k) = cu*matmult(x1,ones);
```

der $*$ betyr elementvis multiplikasjon, lager derfor en matrise hvor de ettårige befolkningstallene etter utdanning blir 'fordelt' til riktig posisjon i henhold til den flerårige aldersgruppeinndelingen. Denne matrisen $nhu\&(j)\&(k)$ blir så dividert elementvis med 118×14 matrisen laget av variablene $fler\&(j)\&(k)$, og lagret på matrisen $ggu\&(j)\&(k)$. Den ettårige aldersinndelingen av befolkningen etter utdanning, kan nå bestemmes ved å matrisemultiplisere fordelingsmatrisen $ggu\&(j)\&(k)$ med den flerårige befolkningen etter utdanning, lagret i matrisene $khat\&(j)\&(k)$:

```
>>dofile bhat\&(j)\&(k) = matmult(ggu\&(j)\&(k),khat\&(j)\&(k));
```

De ettårige befolkningstallene etter utdanning, $bhat\&(j)$ blir så aggregert til variabelen $buhat$:

```
buhat = bhat1 + bhat2 + bhat3
```

Når vi så setter

```
bhat\&(j) = (bhat\&(j)/buhat)*bhat\&(aar1)
```

beregnes derfor en ettårig befolkning etter utdanning, med samme utdanningsfordeling som den eksogent beregnede flerårige befolkningen fra ARBEID, og slik at befolkningen er konsistent med den totale ettårige befolkningen (rendemografisk fremskrivning) ved inngangen til året; $bhat\&(aar1)$.

Deretter hentes de endogent beregnede tallene for inn- og utvandring (etter kjønn, alder og region) fra ARBEID; variablene $inv\&(i)\&(1)\&(k)$ og $utv\&(i)\&(1)\&(k)$. På samme måte som den flerårige befolkningen etter utdanning, $butg$, blir tallene for inn- og utvandring slått sammen til matriser, henholdsvis $invx$ og utv . Netto innvandring etter flerårig alder, $ninv$, blir så beregnet som en differanse av disse. Tallene for netto innvandring blir så splittet opp til ettårig struktur, ved at vi antar at netto innvandring har samme ettårige struktur som befolkningen. Vi bruker nok en gang forholdet mellom $bhat\&(j)$ og $buhat$ til å beregne ettårig netto innvandring etter utdanning (antar at innvandringen har samme utdanningsfordeling som befolkningen). Til slutt blir både den ettårige totale befolkningen, og den ettårige befolkningen etter utdanning oppdatert med netto ettårig innvandring:

```
>>dofile nhat&(aar) = bhat&(aar-1) + ninv&(aar);
>>dofile nhat&(j) = bhat&(j) + ninvu&(j);
```

3.2 Macroen flytt

Macroen `flytt` korrigerer fremskrivningen ovenfor med innenlandske flyttinger, og beregner dermed fremskrivningen for inngangen av året. Gangen er omtrent som følger:

1. Først hentes endogene tall for innenlandske flyttinger etter flerårig alder fra ARBEID, $in&(i)&(l)&(j)&(k)$ og $ut&(i)&(l)&(j)&(k)$, og slås sammen til matrisene $infe&(j)&(k)$ og $utfe&(j)&(k)$. Netto flytting blir beregnet som en differanse av disse.
2. Deretter splittes netto flytting opp etter ettårige aldersgrupper ved hjelp av fordelingsmatrisen $fv&(j)$, og befolkningen blir oppdatert med disse tallene. Total netto flytting for gitt personkjenneegn (kjønn, alder og utdanning) blir eventuelt korrigert slik at den blir lik 0.
3. Vi regner også ut tallene for flytting av barn $nfb&(i)&(k)$ ved hjelp av matrisen $cbarn$, og fordelingsvektoren $flb&(i)$. Også disse tallene blir korrigert slik at netto flytting etter gitt personkjenneegn er lik 0.
4. Til slutt oppdateres befolkningsvariablene $bef&(aar)$ og $bef&(j)&(aar)$ med tallene for netto flytting.

Som nevnt over starter macroen `flytt` ved at endogent beregnede tall for flytting etter flerårig alder og utdanning fra ARBEID hentes inn. På samme måte som tallene for inn- og utvandring i macroen `ninv` slås flyttetallene sammen til matriser, og netto flytting (etter flerårig alder og utdanning) blir beregnet som en differanse av disse. Disse nettotallene lagres på variablene $nfke&(j)&(k)$. Vi har en fast fordelingsmatrise (118×14) $fv&(j)$, for å splitte opp netto flyttetall til ettårig struktur. Dette gjøres derfor med kommandoen:

```
>>dofile nfb&(j)&(k) = matmult(fv&(j), nfke&(j)&(k));
```

118×1 vektorene $nfb&(j)&(k)$, som inneholder de regionvise nettoflyttetallene etter utdanning og etter ettårig alder, blir slått sammen over region til 118×7 matriser $nfb&(j)&(aar)$. Deretter korrigeres flyttetallene slik at totale netto flyttetall for hver ettårige aldersgruppe etter utdanning er lik null. Dette gjøres ved å først regne ut nettoflyttingen totalt. Dersom denne er forskjellig fra null, fordeler vi nettotallene på alle regionene (ved å dividere med sju), og deretter trekker vi dette avviket i fra netto flyttetall for de ulike regionene:

$$nfb_{j,k} = nfb_{j,k} - \left(\sum_{k=1}^7 nfb_{j,k} \right) / 7$$

Vi ser at dersom total netto flytting er positiv trekker vi i fra en positiv størrelse, dvs vi trekker den totale netto flyttingen ned til null. Dersom total netto flytting i utgangspunktet er negativ trekker vi i fra en negativ størrelse, dvs vi justerer netto flytting *opp* til null. Den ettårige befolkningen etter utdanning blir så fremskrevet med kommandoen:

```
>>dofile bef&(j)&(aar) = nhat&(j) + nfb&(j)&(aar);
```

der vi husker at $nhat&(j)$ var den ettårige fremskrivningen av befolkningen etter utdanning korrigert med netto innvandring i macroen *ninv*. Netto flytting etter de tre utdanningskategoriene blir så lagt sammen til ettårig netto flytting for voksne, dvs de som er i alderen 16 – 74 år. Disse slås så sammen med tallene for barns flyttinger hentet fra ARBEID, og splittet opp etter ettårig struktur, samt at vi setter de eldres flyttinger lik null. Dermed får vi dannet en matrise $nfb&(aar)$ med totale netto flyttetall etter ettårig alder for hele befolkningen. Vi avslutter macroen *flytt* ved å oppdatere befolkningen med disse tallene:

```
>>dofile bef&(aar) = nhat&(aar) + nfb&(aar);
```

der $nhat&(aar)$ er den midlertidig fremskrevne ettårige befolkningen inkludert innvandring beregnet i macroen *ninv*.

3.3 Macroen newborn

Denne macroen aldrer befolkningen ved inngangen av året slik at den får alderskjennetegn som ved utgangen av året (dvs dødelighet tas ikke med her). Macroen legger også inn en ny kohort med 0-åringer ved å regne ut antall nyfødte gutter og jenter. Macroen har denne strukturen:

1. Først aldrer befolkningen ved at alle aldersgrupper blir flyttet en plass frem i vektoren (alle er blitt ett år eldre). De nyfødte blir plassert først, mens siste aldersgruppe (85+) blir oppdatert ved å legge sammen tallene for de som var 85+ sist år med de som var 84. Dersom simuleringsåret er 1995 eller tidligere erstattes de fremskrevne befolkningstallene med historiske tall.
2. Deretter beregnes en ny kohort 0-åringer ved å multiplisere befolkningen med fødselsratene *fbm* og *fbk*. Det antas at det kun er kvinner i alderen 15 – 44 år som føder barn. De andre ratene er derfor lik 0. Dermed har vi en befolkning som ved inngangen til året (eventuelt inkludert innvandring og flytting), men kjennetegnet med alderen ved utgangen.

Matrisene *fbm* og *fbk* har dimensjon 86×7 , og inneholder altså regionvise fødselsrater for henholdsvis menn og kvinner. Som nevnt over antar vi at det er kun kvinner i aldersgruppen 15 – 44 år som føder barn, og derfor er alle rater untatt de som ligger i rad 15 – 44 i matrisene *fbm* og *fbk* lik null. Strengt talt korresponderer dette med de kvinnene som er 14 – 43 år gamle ved *inngangen* til (simulerings)året, men når disse blir aldrer blir de 15 – 44. Vi plukker ut de riktige radene fra matrisene over ved hjelp av hjelpevariabelen *bm* som inneholder indeksene 15 – 44, og submatkommandoen i TROLL:

```
>>do fbmc&(k) = submat(fbm,bm,col(fbm)==&(k));
```

Kommandoen over er lagt inn i en for-løkke over region, slik at vi splitter opp tallene i regionvise vektorer, samtidig som vi plukker ut de ratene som er forskjellig fra null. Deretter multipliseres ratene med den kvinnelige befolkningen i alderen 15 – 44 år (ved utgangen av året):

```
>>do fm&(k) = matmult(transp(fbmc&(k)),submat(bef&(aar),bk,&(k)));
```

Her bruker vi hjelpevariabelen *bk* som inneholder indeksene for kvinner i alderen

15 – 44 i befolkningsmatrisen `bef&(aar)`. Eksemplene over viser hvordan vi beregner antall nyfødte gutter; `fm&(k)`. Antall nyfødte jenter beregnes helt analogt, og lagres i variablene `fk&(k)`.

Neste step er å kopiere befolkningsmatrisen `bef&(aar)` til en lokal variabel `bb&(aar)`. Denne variabelen `bb` blir så oppdatert ved at vi først putter inn nyfødte gutter og jenter i henholdsvis rad 1 og rad 88 (husk på at kohortene nå blir forskjøvet og at vi har en kohort ekstra for hvert kjønn. Guttene er derfor i rad 1 – 87 og kvinnene i rad 88 – 174):

```
>>dofile bb&(aar) = addrow(bb&(aar),0,fm);
```

Dermed har vi fått aldret befolkningen ved at alle kohorter er forskjøvet en rad frem i matrisen. Neste step er å legge sammen henholdsvis rad 87 og 88 og rad 173 og 174. Dette er kohortene med de som er 84 og 85+ ved inngangen til året. Disse skal nå tilhøre samme kohort; 85+, og de må derfor slås sammen:

```
>>do ga1 = submat(bb&(aar),86) + submat(bb&(aar),87);
```

Deretter brukes `setrep`-kommandoen i TROLL til å putte de nylig beregnede 85+ kohortene på riktig plass i befolkningsmatrisen `bb&(aar)`:

```
>>dofile bb&(aar) = setrep(bb&(aar),ga1,86,rad);
```

der rad inneholder indeksene fra 1 – 7 og forteller oss at alle kolonner (regioner) skal inkluderes. Til slutt må de nå overflødige linjene 87 og 174 fjernes fra matrisen `bb&(aar)`. Dette gjøres med `delrow`-kommandoen i TROLL:

```
>>dofile bb&(aar) = delrow(bb&(aar),87);
```

Macroen `newborn` avsluttes med å aldre befolkningen etter utdanning også. Det er de som er 15 – 73 år gamle ved inngangen til året vi skal ha med videre. De som er 15 år gamle ved inngangen til året har alle laveste utdanning (kategori 1). Derfor legges hele denne kohorten inn i rad 1 i matrisen med befolkning etter laveste utdanning. Tilsvarende kohorter med utdanning 2 og 2 settes derfor lik null:

```
>>do befut1&(aar) = addrow(bef1&(aar),0,submat(bb&(aar),17));
```

```
>>do befut2&(aar) = addrow(bef2&(aar),0,0);
```

```
>>do befut3&(aar) = addrow(bef3&(aar),0,0);
```

Vi ser at de andre kohortene er blitt forskjøvet en plass, og dermed aldret ett år. Derfor er disse nå kjennetegnet med alderen ved *utgangen* av (simulerings)året. Befolkningen etter utdanning kompletteres ved å skyve ut de som blir 75 år i løpet av året. Dette gjøres ved hjelp av `delrow`-kommandoen. Det er viktig at vi ikke inkluderer dødelighet i tallene produsert av denne macroen. Hovedgrunnen til dette er at disse tallene skal brukes i modellen ARBEID blant annet til å beregne flytting. Flyttingen skal beregnes på grunnlag av befolkningen ved inngangen av året (men kjennetegnet med alder ved utgangen). Det korrekte er derfor å aldre befolkningen som vi har gjort, men IKKE inkludere dødeligheten.

3.4 Macroen `innbef`

Macroen `innbef` lager eksogene variable til bruk i modellen ARBEID. Gangen er som følger:

1. De ettårige befolkningsfremskrivningene fra `newborn` aggregeres først til REGARDSs flerårige aldersgrupper.
2. Deretter lages diverse variable for bruk i neste runde av ARBEID:
`b&(i)&(1)&(j)&(k)`, `b&(i)&(1)&(k)`, `bm&(i)33&(k)`, og `barn&(i)&(1)&(k)`.

Variabelen `bm&(i)33&(k)` er en spesiell befolkningsvariabel for de i aldersgruppe 3 (16 – 19 år) med høyere utdanning. Som nevnt over starter macroen `innbef` med å hente inn de ettårige befolkningsfremskrivningene fra `newborn`; `bb&(aar)` og `befut&(j)&(aar)`. Disse aggregeres til REGARDSs flerårige aldersgrupper ved hjelp av matrisen `cc`:

```
>>do k&(aar) = matmult(transp(cc),bb&(aar));
```

Denne matrisen splittes så i to (etter kjønn) i matisene `k1&(aar)` og `k2&(aar)`. Disse matrisene splittes så opp i enkeltvariable, og gjøres om til tidsserier ved hjelp av `overlay`- og `reshape`-kommandoene i TROLL:

```
>>dofile b&(i)&(1)&(k) =  
>>overlay(reshape(submat(k&(i)&(aar),&(1),&(k)),&(aar1)),  
>>b&(i)&(1)&(k));
```

der kommandoen er plassert i en trippel `for`-løkke over kjønn (`i`), alder (`1`) og region (`k`). Tidsserien `b` eksisterer fra før, og `overlay`-kommandoen sørger for at denne blir oppdatert ved at en ny observasjon blir lagt til. Denne nye observasjonen er nettopp det korresponderende elementet fra vår flerårige befolkningsmatrise `k&(i)&(aar)`. Dette elementet gis *tidsseriedimensjon* ved hjelp av kommandoen `reshape`. Resultatet av operasjonene over er altså at vi har konvertert våre flerårige befolkningsmatriser til tidsserier, som så kan brukes som eksogene variable i modellen ARBEID. De andre variablene (tidsseriene) produsert av denne macroen (som nevnt over) blir laget på en helt analog måte.

3.5 Macroen `rb`

Macroen `rb` henter befolkningen beregnet av `newborn`, og lager en rendemografisk fremskrivning for utgangen av året ved å inkludere dødeligheten. Dette gjøres ved å multiplisere befolkningen (hentet fra `newborn`) med overlevelsesratene `sm` og `sk` for hhv menn og kvinner:

```
>>dofile bh&(aar) = ol&(aar)*bb&(aar);
```

Ved å kjøre `rb` etter `newborn` får vi også beregnet spedbarnsdødeligheten. Den rendemografiske fremskrivningen blir lagret i variabelen `bhat&(aar)`. Denne brukes delvis i den neste macroen (`utbef`) for å beregne eksogene variable til ARBEID, og delvis i den neste runden i demografidelen, blant annet i macroen `ninv`.

3.6 Macroen `utbef`

Den siste macroen som kalles på av `dgraf` er `utbef`. Denne macroen henter den rendemografiske fremskrivningen `bhat&(aar)`, aggregerer til REGARDSs flerårige aldersstruktur, og lager variabelen `butg&(i)&(1)&(k)` som brukes som eksogen variabel i neste runde av ARBEID. Teknisk sett gjøres dette på samme måte som

tilsvarende konstruksjoner i macroen innbef, ved hjelp av kommandoene overlay, reshape og submat i TROLL.

DEL VI

Eksport av resultater

1 Innledning

I denne delen skal vi se nærmere på de rutinene som brukes for å eksportere resultattall fra REGARD. Eksport av resultatene er den siste operasjonen som gjøres i styringsmacroen `regard`, og denne utføres ved et kall på macroen `export`. Tekstfilene med resultattall blir lagret på underkatalogen `regard`. Resultattallene blir også lagret som TROLL-matriser (.dat-filer), og disse, samt endel hjelpematriser blir lagret på underkatalogen `res`. Følgende navnestruktur gjelder for resultatmatrisene:

- `beftot.txt` - Totale befolkningstall etter region, kjønn og ettårig alder.
- `befutd1.txt`, `befutd2.txt`, `befutd3.txt` - Befolkning etter kjønn, utdanning, region og ettårig alder.
- `nfbtot.txt` - totale netto flyttetall etter region, kjønn og ettårig alder.
- `nfbutd1.txt`, `nfbutd2.txt`, `nfbutd3.txt` - Netto flytting etter utdanning, kjønn, region og ettårig alder.
- `arbst1.txt`, `arbst2.txt`, `arbst3.txt` - Arbeidsstyrke etter utdanning, region, kjønn og (flerårige) aldersgrupper.
- `sysjr.txt` - (Bostads-)sysselsetting etter sektor og region.
- `xm1.txt` - Markedsleie etter kjønn, utdanning og region.

Disse tekstfilene kan så overføres til PC eller stormaskin for videre analyser på regneark eller i SAS. Filene kan også analyseres i SAS på UNIX.

2 Macroer

Eksport av resultater utføres ved å kjøre macroen `export`. Parametrene til macroen `export` er henholdsvis navn på kjøringen, samt første og siste simuleringsår. De siste parametrene trengs (naturligvis) for å vite hvor mye av tidsseriene på resultatdatabasen som skal presenteres. Macroen `export` kaller på fire undermacroer; `exbef`, `arbst`, `sysjr` og `xm1`. Disse macroene leser inn tidsserier fra resultatdatabasen, konverterer disse til tekstformat, og setter de opp på en bestemt form. Vi skal her se nærmere på hvordan disse macroene fungerer.

2.1 Macroen `exbef`

Macroen `exbef` lager befolkningstall og flyttetall. Matrisen `beftot` inneholder totale befolkningstall etter år, kjønn og region. `befutd` inneholder befolkningstallene etter utdanning. Tilsvarende inneholder `nfbtot` og `nfbutd` henholdsvis totale netto flyttetall og netto flyttetall etter utdanning. Disse resultatmatrisene lages ved hjelp av `partcomb`- og `combine`-kommandoene i TROLL. Første kolonne i den produserte matrisen inneholder årstallene, kolonne to inneholder kjønnsindikatoren (1 =menn, 2 =kvinner), kolonne tre alderen (ettårig). De resterende sju kolonnene inneholder befolknings- eller flyttetall for de sju regionene.

Disse tekstmatrisene lages ved å slå sammen (*konkatenerer*) vektorer og matriser ved

hjelp av `partcomb`-kommandoen til større matriser. Hvis `ar` er en vektor som inneholder årstall, `kj` en matrise som inneholder kjønnsindikatorerne og alderen, mens `bef&(i)` er total befolkning i år `i` (etter region), så lages delmatrisen for dette året med følgende kommando:

```
>>dofile b&(i) = partcomb(1,3,ar,kj,bef&(i));
```

Tilsvarende konstrueres også matrisene med flyttetall. Det lages midlertidig en matrise for hvert år, som så slås sammen (settes oppå hverandre) ved hjelp av `partcomb`-kommandoen. For befolknings- og flyttetall etter utdanning lages det en slik tekstmatrise for hver utdanningskategori. Matrisene eksporteres til tekstfiler ved hjelp av følgende sett TROLL-kommandoer:

```
>>do hfopen("beftot.txt","w");
>>do fps = hfogetfp("beftot.txt");
>>do hfprintf(fps,"%5.0f ..... \n", var1,var2,...);
>>do hfclose(fps);
```

Vi bruker altså TROLL-kommandoene `hfopen`, `hfogetfp`, `hfprintf` og `hfclose` for å handtere rene tekst-filer. Parameter-strengen i `hfprintf` er som følger: `fps` er filpekeren (definert ved hjelp av `hfogetfp`-kommandoen), den kryptiske strengen inne i hermetegn (`%5.0f .. osv`) angir formatet på utskriften, på samme måte som i C eller FORTRAN. Formatet `%5.0f` betyr at tallet skal skrives ut som et flyttall (desimaltall) (derav 'f') over fem plasser, og med null plasser etter desimalpunktumet. Dette er det samme som å skrive ut et helt tall over fem plasser. Avslutningen `\n` betyr at det skal foretas et linjeskift før neste variabel skrives ut på fila. Den avsluttende oppramsingen av `variable; var1, var2,` er navnene på de variablene som skal skrives ut.

2.2 Macroen `arbst`

Denne macroen lager tallene for arbeidsstyrke. Det lages tre matriser, en for hver utdanningskategori. Hver matrise inneholder arbeidsstyrketall etter år, kjønn og region. Virkemåten er svært lik den foregående macroen. Den eneste reelle forskjellen er at matrisen med arbeidsstyrketall etter region må lages. Dette gjøres ved at vi (ved hjelp av `values`-kommandoen) henter ut de beregnede tallene for arbeidsstyrke (som ligger lagret som tidsserier), og slår disse sammen til matriser (ved hjelp av kommandoene `combine` og `addrow`). Dette gjøres først i macroen `arbst`. Vi lagrer arbeidsstyrketallene midlertidig på delmatriser `as&(i)&(j)&(a)` etter kjønn, utdanning og årstall. På samme måte som i foregående macro slås disse matrisene sammen til større matriser ved hjelp av `partcomb`-kommandoen. Til slutt kopieres matrisene til tekstfiler på samme måte som i `exbef`.

2.3 Macroen `sysjr`

Denne macroen produserer sysselsettingstall etter år, sektor og region. På samme måte som i macroen `arbst` hentes beregnede sysselsettingstall (tidsserier) ved hjelp av `values`-kommandoen i TROLL, og en matrise `ljr` av sysselsettingstall etter sektor og region (bruker nok en gang kommandoene `partcomb`, `combine` og `addrow`) blir konstatert. Det eneste tekniske problemet oppstår når vi skal lage en kolonne med sektorkoder i tekstmatrisen som skal skrives ut. Sektorkodene for offentlig

sektor inneholder både tall og bokstaver (f.eks. '92S'), noe som gjør at TROLL splitter den i to. Dette har vi løst ved å representere '92S' som tallet '921', '92K' er lik '922' osv.

2.4 Macroen `xml`

Macroen `xml` produserer resultattall for markedsleien, etter år, kjønn, utdanning og region. På samme måte som i de foregående macroene henter vi inn beregnede tall for markedsleie (tidsserier) ved hjelp av `values`-kommandoen, og bygger opp en matrise `xml` av markedsleietall etter år og kjønn (bruker nok en gang kommandoene `partcomb`, `combine` og `addrow`). Til slutt eksporteres matrisen `xml` til en tekstfil etter samme mønster som i de foregående macroene.

Appendix A

Lister og aggregeringer i REGARD

1 Aldersgruppene i REGARD

REGARD har i dag en *flerårig aldersgruppeinndeling*, det vil si at befolkningen er delt inn i et visst antall aldersklasser, med flere ettårige aldersgrupper i hver klasse. I REGARD har vi delt inn befolkningen i elleve aldersgrupper (se Mohn et.al. [1994]):

1. 0 – 6 år.
2. 7 – 15 år.
3. 16 – 19 år.
4. 20 – 24 år.
5. 25 – 34 år.
6. 35 – 44 år.
7. 45 – 59 år.
8. 60 – 66 år.
9. 67 – 74 år.
10. 75 – 84 år.
11. 85 år og over.

Barn og unge er altså aldersgruppe 1 og 2. De yrkesaktive voksne er aldersgruppene 3-9 (16-74 år), mens eldre (over 75 år) er aldersgruppe 10 og 11.

2 Sektorer i MODAG/KVARTS. Aggregering fra FNR

Sektorinndelingen i REGARD er den samme som i MODAG/KVARTS. Inndelingen, består av 28 sektorer, og er som følger (se også f.eks. Mohn et.al. [1994]):

1. 11 - Jordbruk.
2. 12 - Skogbruk.
3. 13 - Fiske og fangst.
4. 14 - Fiskeoppdrett.
5. 15 - Produksjon av konsumvarer.
6. 25 - Produksjon av vareinnsats og investeringsvarer.
7. 34 - Produksjon av treforedlingsprodukter.
8. 37 - Produksjon av kjemiske råvarer.
9. 40 - Raffinering av råolje.
10. 43 - Produksjon av metaller.
11. 45 - Produksjon av verkstedsprodukter.
12. 50 - Produksjon av skip og oljeplattformer.
13. 71 - Elektrisitetsproduksjon.
14. 55 - Bygge- og anleggsvirksomhet.
15. 81 - Varehandel.
16. 64 - Utvinning og transport av råolje og naturgass.
17. 65 - Utenriks sjøfart og oljeboring.
18. 74 - Innenriks samferdsel.
19. 63 - Bank og forsikringsvirksomhet.
20. 83 - Boligtjenester.
21. 85 - Annen privat tjenesteproduksjon.
22. 92S - Forsvar.
23. 93S - Statlig undervisning og forskningsvirksomhet.
24. 94S - Statlige helsetjenester.
25. 95S - Annen statlig tjenesteproduksjon.
26. 93K - Kommunal undervisning og forskningsvirksomhet.
27. 94K - Kommunale helsetjenester.
28. 95K - Annen kommunal tjenesteproduksjon.

2.1 Aggregering fra fylkesvist nasjonalregnskap (FNR)

Når vi skal beregne andeler for å foreta fylkesvis oppsplitting av tall fra MODAG eller KVARTS, bruker vi, som nevnt i del II, tall fra det fylkesvise nasjonalregnskapet (FNR). FNR-tallene er lagret på såkalt NR-nivå, og for å kunne bruke FNR-tallene må de derfor aggregeres til MODAG/KVARTS-nivå. Oversiktene under viser sammenhengen mellom sektorkoder på NR-nivå og ditto på MODAG-nivå, for henholdsvis bruttoprodukt (produksjon), investering og sysselsetting. Spesifikasjonene under er implementert i macroene `afratilps`, `afratiljs` og `afratilsy`. Se også utskriften av kildekoden av disse macroene.

Bruttoprodukt

$$11 = 22010 + 22015 + 23010 + 23014$$

$$12 = 23020$$

$$13 = 22051 + 23051$$

$$14 = 23052$$

$$15 = 23151 + 23152 + 23153 + 23154 + 23155 + 23156 + 23157 + 23158 + 23159 + 23160 + 23170 + 23180 + 23190$$

$$25 = 23100 + 23130 + 23140 + 23201 + 23202 + 23203 + 23204 + 23221 + 23222 + 23243 + 23244 + 23245 + 23246 + 23250 + 23265 + 23261 + 23262 + 23266 + 23361 + 23362 + 23363 + 23372$$

$$34 = 23211 + 23212 + 23213$$

$$37 = 23241 + 23242 + 23247$$

$$40 = 23231 + 23232$$

$$43 = 23271 + 23273 + 23274 + 23275$$

$$45 = 23281 + 23286 + 23287 + 23291 + 23293 + 23296 + 23297 + 23300 + 23311 + 23313 + 23314 + 23321 + 23323 + 23331 + 23334 + 23340 + 23353 + 23354 + 23355$$

$$50 = 23351 + 23352$$

$$71 = 23401 + 23402 + 23403$$

$$55 = 22452 + 23451 + 23452 + 23453 + 23454$$

$$81 = 23501 + 23505 + 23510 + 23521$$

$$64 = 23111 + 23608$$

$$65 = 23112 + 23611$$

$$74 = 23601 + 23603 + 23605 + 23604 + 23602 + 23613 + 23620 + 23631 + 23632 + 23633 + 23641 + 23642$$

$$63 = 23651 + 23652 + 23655 + 23661 + 23662 + 23663 + 23670$$

83 = 22704

85 = 22950 + 23405 + 23551 + 23553 + 23502 + 23527 + 23700 + 23711
+ 23713 + 23720 + 23730 + 23741 + 23742 + 23744 + 23745 + 23747 +
23748 + 23800 + 23852 + 23851 + 23853 + 23859 + 23900 + 23910 + 23921
+ 23922 + 23926 + 23927 + 23930 + 26800 + 26851 + 26853 + 26854 +
26910 + 26921 + 26926

92S = 24752

93S = 24800

94S = 24852 + 24851 + 24853

95S = 24601 + 24631 + 24632 + 24670 + 24730 + 24742 + 24745 + 24751
+ 24921

93K = 25800

94K = 25851 + 25853 + 25854

95K = 25410 + 25453 + 25751 + 25900 + 25921

Investering

11 = 83010

12 = 83020

13 = 83051

14 = 83052

15 = 83151 + 83152 + 83153 + 83154 + 83155 + 83156 + 83157 + 83158
+ 83159 + 83160 + 83170 + 83180 + 83190

25 = 83100 + 83130 + 83140 + 83201 + 83202 + 83203 + 83204 + 83221
+ 83222 + 83243 + 83244 + 83245 + 83246 + 83250 + 83265 + 83261 +
83262 + 83266 + 83361 + 83362 + 83363

34 = 83211 + 83212 + 83213

37 = 83241 + 83242 + 83247

40 = 83231 + 83232

43 = 83271 + 83273 + 83274 + 83275

45 = 83281 + 83286 + 83287 + 83291 + 83293 + 83296 + 83297 + 83300
+ 83311 + 83313 + 83314 + 83321 + 83323 + 83331 + 83334 + 83340 +
83353 + 83354 + 83355

50 = 83351 + 83352

71 = 83401 + 83402 + 83403

$$55 = 83451 + 83452 + 83453 + 83454$$

$$81 = 83501 + 83505 + 83510 + 83521$$

$$64 = 83111 + 83608$$

$$65 = 83112 + 83611$$

$$74 = 83601 + 83603 + 83605 + 83604 + 83602 + 83613 + 83620 + 83631 + 83632 + 83633 + 83641 + 83642$$

$$63 = 83651 + 83652 + 83655 + 83661 + 83662 + 83663$$

$$83 = 82704$$

$$85 = 83405 + 83551 + 83553 + 83502 + 83527 + 83700 + 83711 + 83713 + 83720 + 83730 + 83741 + 83742 + 83744 + 83745 + 83747 + 83748 + 83800 + 83852 + 83851 + 83853 + 83859 + 83900 + 83910 + 83921 + 83922 + 83926 + 83927 + 83930 + 86800 + 86851 + 86853 + 86854 + 86910 + 86921 + 86926$$

$$92S = 84752$$

$$93S = 84800$$

$$94S = 84852 + 84851$$

$$95S = 84601 + 84631 + 84632 + 84730 + 84742 + 84745 + 84751 + 84921$$

$$93K = 85800$$

$$94K = 85851 + 85853 + 85854$$

$$95K = 85410 + 85751 + 85900 + 85921$$

Sysselsetting

$$11 = 23010 + 23014$$

$$12 = 23020$$

$$13 = 23051$$

$$14 = 23052$$

$$15 = 23151 + 23152 + 23153 + 23154 + 23155 + 23156 + 23157 + 23158 + 23159 + 23160 + 23170 + 23180 + 23190$$

$$25 = 23100 + 23130 + 23140 + 23201 + 23202 + 23203 + 23204 + 23221 + 23222 + 23243 + 23244 + 23245 + 23246 + 23250 + 23265 + 23261 + 23262 + 23266 + 23361 + 23362 + 23363 + 23372$$

$$34 = 23211 + 23212 + 23213$$

$$37 = 23241 + 23242 + 23247$$

40 = 23231 + 23232

43 = 23271 + 23273 + 23274 + 23275

45 = 23281 + 23286 + 23287 + 23291 + 23293 + 23296 + 23297 + 23300
+ 23311 + 23313 + 23314 + 23321 + 23323 + 23331 + 23334 + 23340 +
23353 + 23354 + 23355

50 = 23351 + 23352

71 = 23401 + 23402 + 23403

55 = 23451 + 23452 + 23453 + 23454

81 = 23501 + 23505 + 23510 + 23521

64 = 23111 + 23608

65 = 23112 + 23611

74 = 23601 + 23603 + 23605 + 23604 + 23602 + 23613 + 23620 + 23631
+ 23632 + 23633 + 23641 + 23642

63 = 23651 + 23652 + 23655 + 23661 + 23662 + 23663 + 23670

83 = 22704

85 = 22950 + 23405 + 23551 + 23553 + 23502 + 23527 + 23700 + 23711
+ 23713 + 23720 + 23730 + 23741 + 23742 + 23744 + 23745 + 23747 +
23748 + 23800 + 23852 + 23851 + 23853 + 23859 + 23900 + 23910 + 23921
+ 23922 + 23926 + 23927 + 23930 + 26800 + 26851 + 26853 + 26854 +
26910 + 26921 + 26926

92S = 24752

93S = 24800

94S = 24852 + 24851 + 24853

95S = 24631 + 24632 + 24670 + 24730 + 24742 + 24745 + 24751 + 24921

93K = 25800

94K = 25851 + 25853 + 25854

95K = 25410 + 25453 + 25751 + 25900 + 25921

3 Fylkeskoder og regioninndelingen i REGARD

Under følger en oversikt over fylkeskodene i REGARD:

1. 01 - Østfold.
2. 02 - Akershus.
3. 03 - Oslo.

4. 04 - Hedmark.
5. 05 - Oppland.
6. 06 - Buskerud.
7. 07 - Vestfold.
8. 08 - Telemark.
9. 09 - Aust-Agder.
10. 10 - Vest-Agder.
11. 11 - Rogaland.
12. 12 - Hordaland.
13. 14 - Sogn og Fjordane.
14. 15 - Møre og Romsdal.
15. 16 - Sør-Trøndelag.
16. 17 - Nord-Trøndelag.
17. 18 - Nordland.
18. 19 - Troms.
19. 20 - Finnmark.
20. 99 - Ekstrafylket.

I formodellen (del **III**) foretas som nevnt en fylkesvis oppsplitting av tallene for bruttoprodukt, investering og sysselsetting fra MODAG/KVARTS. I modellen ARBEID benyttes REGARDS regioninndeling i sju regioner, gitt i Mohn et.al. [1994]. Før de fylkesvis oppsplittede tall fra MODAG/KVARTS kan brukes i ARBEID må de derfor aggregeres til REGARDS regioninndeling. Under følger en oversikt over de sju regionene, og hvordan sammenhengen mellom disse og den fylkesvise inndelingen er.

1. Oslo og Akershus - Fylke 02 + 03.
 2. Østlandet kyst (Østfold, Buskerud, Vestfold og Telemark) - Fylke 01 + 06 + 07 + 08.
 3. Hedmark/Oppland - Fylke 04 + 05.
 4. Agder og Rogaland - Fylke 09 + 10 + 11.
 5. Vestlandet (Hordaland, Sogn og Fjordane, og Møre og Romsdal) - Fylke 12 + 14 + 15.
 6. Trøndelag - Fylke 16 + 17.
 7. Nord-Norge (Nordland, Troms og Finnmark) - Fylke 18 + 19 + 20.
- Ekstrafylket (99) blir i noen sammenhenger regnet som en åttende region.

Appendix B

Ettårig versjon av REGARD

1 Innledning

Som vi har sett har REGARD i dag en flerårig aldersstruktur. En slik inndeling er en grei forenkling fordi befolkningen deles opp i grupper der hver ettårige kohort i gruppen har omtrent de samme egenskapene. Denne aldersgruppeinndelingen er også hensiktsmessig med tanke på størrelsen på modellen (antall likninger). Der- som vi lager en versjon av REGARD basert på en ettårig aldersgruppeinndeling, vil antall likninger øke med en faktor på omlag åtte (86 istedenfor 11 aldersgrup- per). I gamle TROLL var vi avhengig av denne reduksjonen, for med en ettårig aldersgruppeinndeling ble modellen for stor til at TROLL kunne klare å handtere den. I nye TROLL er vi ikke bundet av slike begrensninger, og det er derfor blitt mer aktuelt å generere en ettårig versjon av REGARD etter at modellen er blitt oversatt til portabel TROLL. En ettårig versjon vil også være mer i tråd med den relaterte *fylkesmodellen*. Nok et argument for å etablere en ettårig modell er at de- mografien er på ettårig format. I en vanlig REGARD-kjøring er vi (som vi allerede har sett) derfor nødt til å splitte opp de flerårige aldersgruppene til ettårige hver gang demografien skal oppdateres. Deretter må vi aggregere tilbake til den flerårige strukturen når modellen skal kjøres. Dette opplegget har flere ulemper:

- Oppsplittingen fra flerårig til ettårig struktur skjer i henhold til fordelingen for året før. Det er ikke gitt at dette er den korrekte fordelingen.
- Oppsplittingen (og aggregeringen) i demografidelen består matematisk sett av en rekke ressurskrevende og tildels obskure matrisemanipulasjoner. Dette er uoversiktlig både for systemansvarlig og bruker. En ettårig modell vil være mye enklere logisk sett.
- Med en kombinasjon av ettårige og flerårige aldersgrupper kan vi kun simulere for ett år av gangen. Mellom hver simulering foretas oppdateringen av de- mografien. Dette samspillet gjør at vi må ut av `simulate`-sesjonen i TROLL. Å starte en simuleringssesjon er en ressurskrevende operasjon, der modellens insidensmatrise sjekkes. Dersom vi kun har ettårige aldersgrupper kan opp- dateringen av demografien inkluderes i modellen. Dermed kan vi simulere for alle årene i én sesjon. Dette vil bety en vesentlig innsparing tidsmessig.

På grunn av de argumenter som er nevnt over vil ikke modellen med ettårige aldersgrupper (som er åtte ganger så stor som den opprinnelige) bli mer tidskrevende

å kjøre. Derfor vil overgangen til en ettårig modell kunne gi oss en gevinst i form av:

- 'Riktigere' resultater (høyere presisjon og bedre konsistens).
- Minst like rask kjøretid.
- Logisk sett en enklere og mer oversiktlig modell.

Våren 1997 ble det derfor bestemt at vi skulle forsøke å utarbeide en versjon av REGARD med ettårige aldersgrupper. Dette appendikset inneholder dokumentasjonen av denne ettårige modellen, slik den foreligger pr. i dag. Det må understrekes at en slik modell krever en god del tilrettelegging, blant annet med å gjøre om befolkningsdata fra matriser til tidsserier for å kunne foreta demografiske oppdateringer under simuleringen. I tillegg må nye sett med koeffisienter, og en ny utgave av eksogene data som er tilpasset den ettårige oppbyggingen estimeres.

2 Macroene

For de macroene som pr. i dag genererer *likninger* til den flerårige versjonen av REGARD, er det ikke store endringer. Den eneste endringen er at løkken for alder (1-løkken) løper fra 1 til 86 istedenfor fra 1 til 11. Det er dessuten slik at ikke alle de tidligere likningsgenererende macroene har noen aldersstruktur. Disse macroene er uendret i den 'ettårige' versjonen av REGARD. For de macroene som inngikk i den tidligere demografidelen er det imidlertid store endringer. For det første er macroene *innbef* og *utbef*, som aggregerte befolkningene til flerårig alder til bruk i modellen (ARBEID), nå blitt overflødige. Dessuten er de andre macroene blitt fundamentalt endret i den forstand at de nå *genererer likninger* fremfor å foreta rene matriseberegninger. For de fleste macroene har dette innebært en betydelig forkortelse. Nedenfor følger en kort beskrivelse av de ulike macroene som genererer demografilikninger. Se også utskriften av kildekoden (som ligger på underkatalogen *ettaar*).

2.1 Macroen *ninv*

Denne macroen genererer likninger som inkluderer netto innvandring i befolkningsstallene. Netto innvandring blir lagret i variabelen *ninv*:

$$\text{ninv}_{i,l,k} = \text{inv}_{i,l,k}(-1) - \text{utv}_{i,l,k}(-1)$$

der indeksen *i* er kjønn, *l* er alder, mens *k* er region, akkurat som tidligere. Befolkningen ved *inngangen til* (simulerings-)året blir oppdatert med netto innvandring:

$$\text{nhat}_{i,l,k} = \text{b}_{i,l,k}(-1) + \text{ninv}_{i,l,k}$$

der $\text{b}_{i,l,k}(-1)$ betyr den laggede versjonen av befolkningsvariabelen $\text{b}_{i,l,k}$, det vil si befolkningen ved *utgangen* av foregående år. For netto innvandring *etter utdanning* utnytter vi den tilhørende endogene variabelen butg_{iljk} i ARBEID (der *j* er utdanningskategorien):

$$\text{ninv}_{i,l,j,k} = \frac{\text{butg}_{i,l,j,k}}{\sum_{m=1}^3 \text{butg}_{i,l,m,k}} \cdot \text{ninv}_{i,l,k}$$

og den tilhørende (midlertidige) befolkningen etter utdanning blir lik:

$$\text{nhat}_{i,l,j,k} = b_{i,l,j,k}(-1) + \text{ninv}_{i,l,j,k}$$

2.2 Macroen flytt

Likningen for netto flytting etter kjønn, alder, utdanning og region er definert som differansen mellom inn- og utflytting:

$$\text{nfl}_{i,l,j,k} = \text{in}_{i,l,j,k} - \text{ut}_{i,l,j,k}$$

Vi definerer *total* netto flytting som summen over regionene, og gjennomsnittet over regionene av denne størrelsen blir derfor lik

$$\text{nfl}_{i,l,j} = \left(\sum_{k=1}^7 \text{nfl}_{i,l,j,k} \right) / 7$$

Som i det flerårige tilfellet skal den totale nettoflyttingen etter ettårig alder være lik null. Dette gjøres ved at vi trekker den gjennomsnittlige netto flyttingen i fra netto flyttingen i hver region. Den *endelige* netto flyttingen etter korrigering blir derfor lik:

$$\text{nfb}_{i,l,j,k} = \text{nfl}_{i,l,j,k} - \text{nfl}_{i,l,j}$$

Den midlertidig oppdaterte befolkningen inkludert flytting (og inn/utvandring) etter kjønn, alder, utdanning og region blir dermed lik

$$\text{bef}_{i,l,j,k} = \text{nhat}_{i,l,j,k} + \text{nfb}_{i,l,j,k}$$

For å finne total netto flytting, *uavhengig av* utdanningsdimensjonen, må vi aggregere:

$$\text{nfb}_{i,l,k} = \text{nfb}_{i,l,1,k} + \text{nfb}_{i,l,2,k} + \text{nfb}_{i,l,3,k}$$

Barnas netto flyttinger er definert ved

$$\text{nfb}_{i,l,k} = \text{inb}_{i,l,k} - \text{utb}_{i,l,k}$$

der de endogene variablene for inn- og utflytting av barn, *inb* og *utb*, er definert i macroen *flytmod5*.

2.3 Macroen newborn

I denne macroen aldres befolkningen, det vil si at alle kohorter forskyves til neste aldersgruppe:

$$\text{bb}_{i,l,k} = \text{bef}_{i,l-1,k}$$

der $l \in \{1, \dots, 84\}$, og $\text{bef}_{i,l,k}$ er den midlertidige befolkningen etter inn- og utvandring og flytting som beregnet av macroene *ninv* og *flytt*. Den eldste aldersgruppen, dvs alder 85+, er definert som summen av de som var 85 år eller eldre i fjor pluss de som blir 85 i år:

$$\text{bb}_{i,85,k} = \text{bef}_{i,85,k} + \text{bef}_{i,84,k}$$

Etter at befolkningen er aldret må nyfødte inkluderes. Dette gjøres ved å multiplisere kvinner i alder 15 til 44 år med *fødselsratene*. For nyfødte gutter (kjønn = 1) har vi:

$$bb_{1,0,k} = \sum_{l=15}^{44} fbm_{l,k} \cdot bb_{2,l,k}$$

og tilsvarende for nyfødte jenter (kjønn = 2):

$$bb_{2,0,k} = \sum_{l=15}^{44} fbk_{l,k} \cdot bb_{2,l,k}$$

For befolkningen etter utdanning (aldersgrupper 16 – 74 år), foretas oppdateringen i to trinn. Vi antar at alle 16-åringene har laveste utdanning (utdanning = 1). Variabelen $befut_{i,16,1,k}$ settes derfor lik $bef_{i,16,k}$ i den midlertidig aldrede befolkningen totalt. De to variablene $befut_{i,16,2,k}$ og $befut_{i,16,3,k}$ settes derfor lik null. For de som i løpet av året blir mellom 17 og 74 år antar vi at den nye midlertidige befolkningen har den samme utdanningsfordelingen som variabelen $butg_{i,l,j,k}$. Denne variabelen inneholder den midlertidig aldrede befolkningen etter utdanning for disse aldersgruppene ved utgangen av forrige år, inkludert eventuelle korrigeringer med tall fra MOSART. Vi har dermed et liknende opplegg som i macroen *ninv*:

$$befut_{i,l,j,k} = \frac{butg_{i,l,j,k}}{\sum_{m=1}^3 butg_{i,l,m,k}} \cdot bb_{i,l,k}$$

Fordelingen er med andre ord som bestemt av variabelen $butg_{i,l,j,k}$, mens totaltallet over utdanning stemmer med den nylig midlertidig aldrede befolkningen etter kjønn, alder og region; $bb_{i,l,k}$.

2.4 Macroen *rb*

Denne macroen inkluderer dødelighet i den midlertidig fremskrevne befolkningen, slik at vi får beregnet *endelig* fremskrevet befolkning ved utgangen av simuleringsåret:

$$b_{i,l,k} = ol_{i,l,k} \cdot bb_{i,l,k}$$

Den endelige fremskrevne befolkningen *etter utdanning* får vi ved å atter bruke tilsvarende opplegg som i *ninv* og *newborn*:

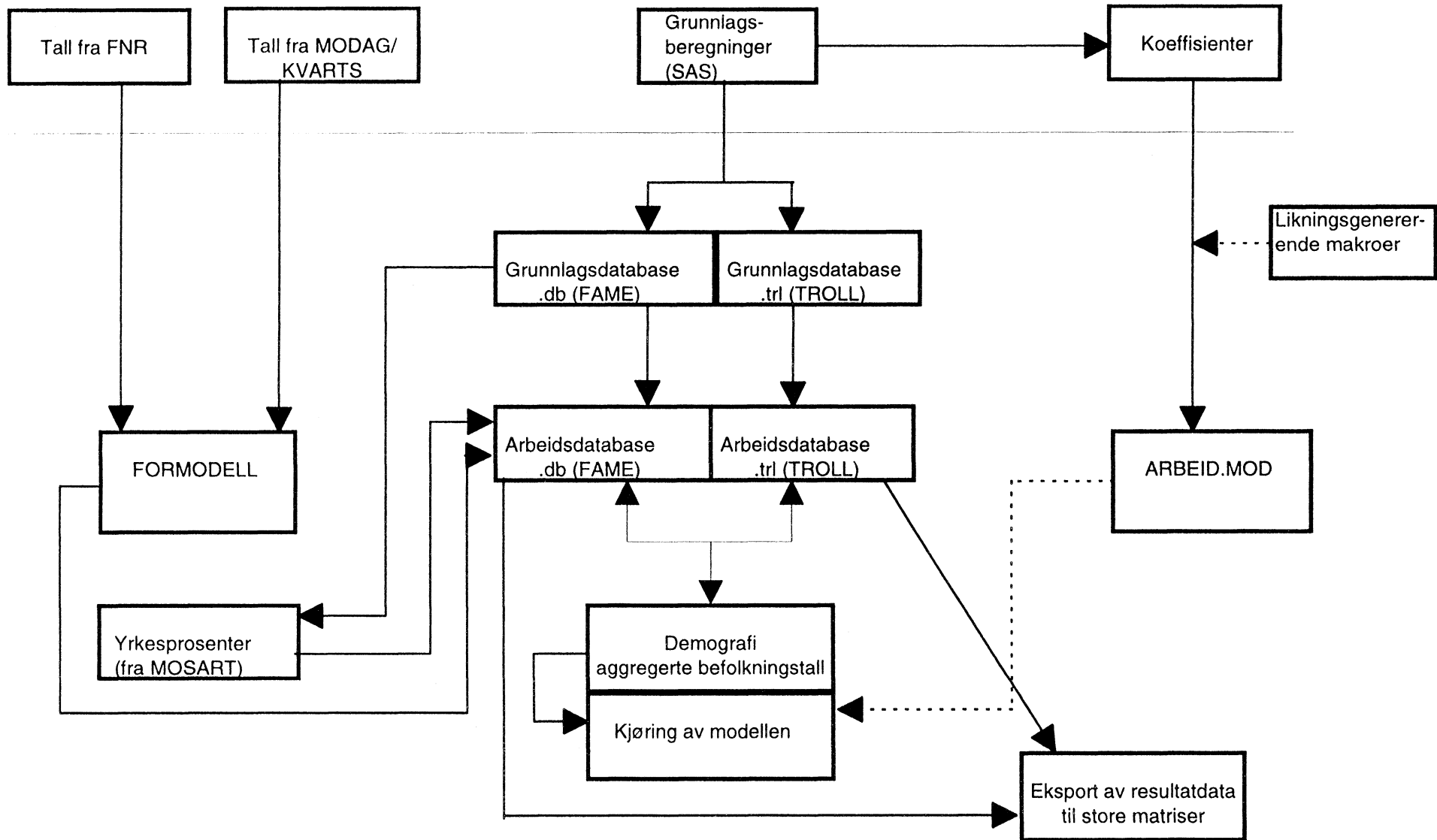
$$b_{i,l,j,k} = \frac{butg_{i,l,j,k}}{\sum_{m=1}^3 butg_{i,l,m,k}} \cdot b_{i,l,k}$$

Denne befolkningen brukes så i de andre likningene i modellen til å beregne ulike variable som sysselsetting, arbeidsstyrke etc.

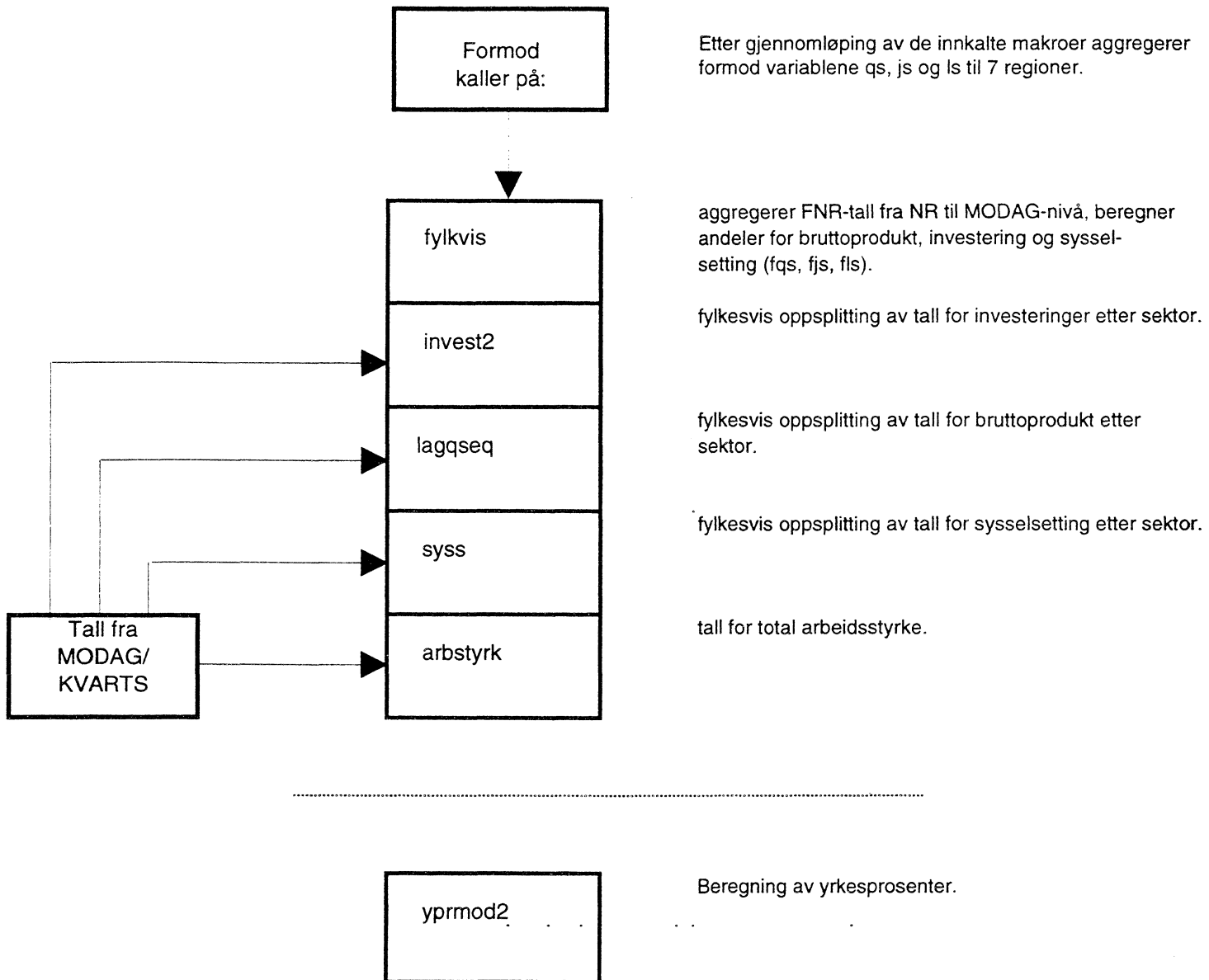
Litteratur

1. Bowitz E. & Holm I. 1993. *Modag - Teknisk Dokumentasjon pr.1.6.93*.
2. Hollinger P. & Spivakovsky L. 1996. *Portable Troll - Reference Manual, release 1.0*. Intex Solutions Inc.
3. Mohn K., Stambøl L.S. & Sørensen K. 1994. *Regional analyse av arbeidsmarked og demografi - Drivkrefter og utviklingstrekk belyst ved modellsystemet REGARD*. Sosiale og økonomiske studier.
4. Mohn K. 1994. *Regard på Troll - En kort introduksjon*.

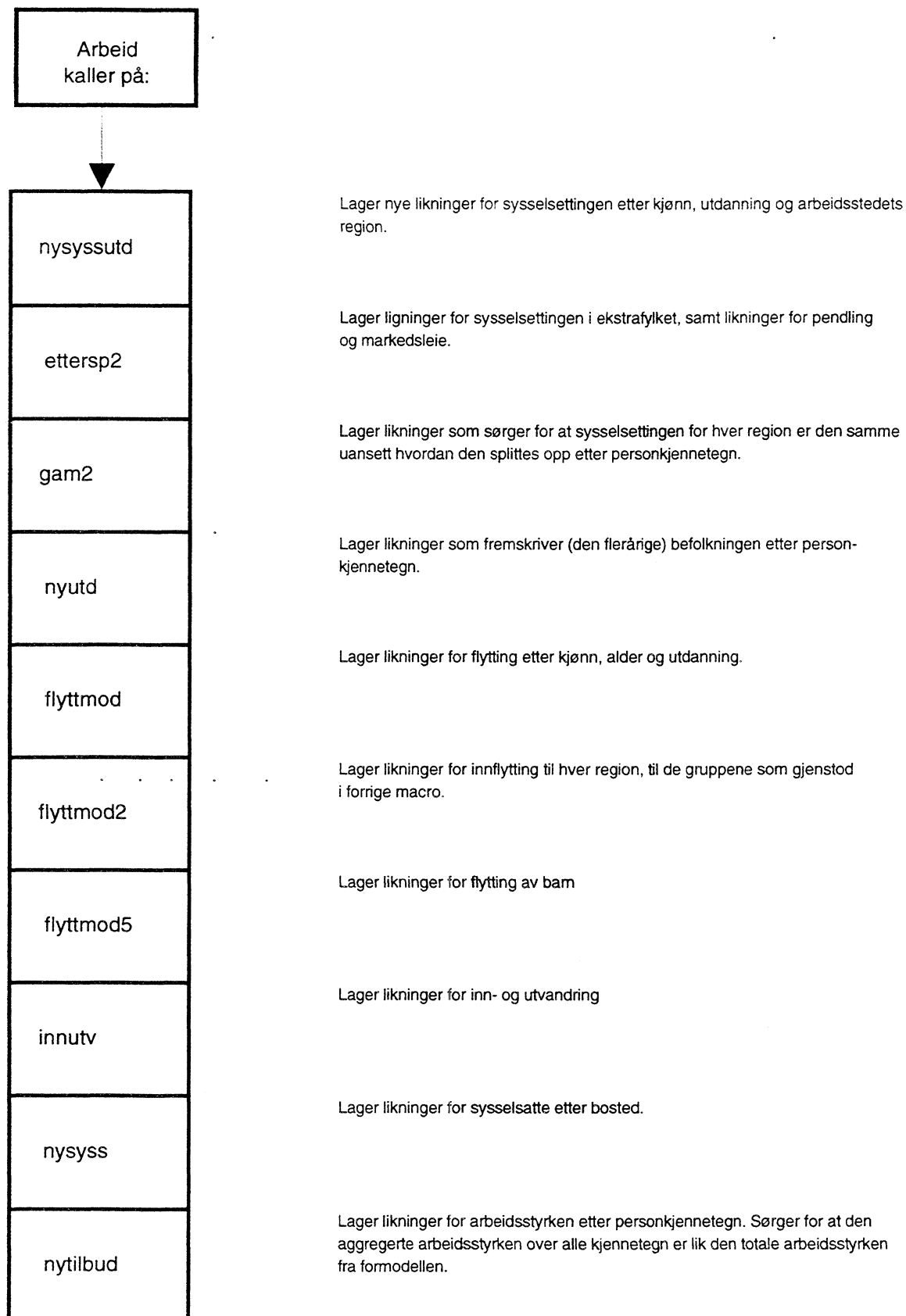
Oversikt over REGARD



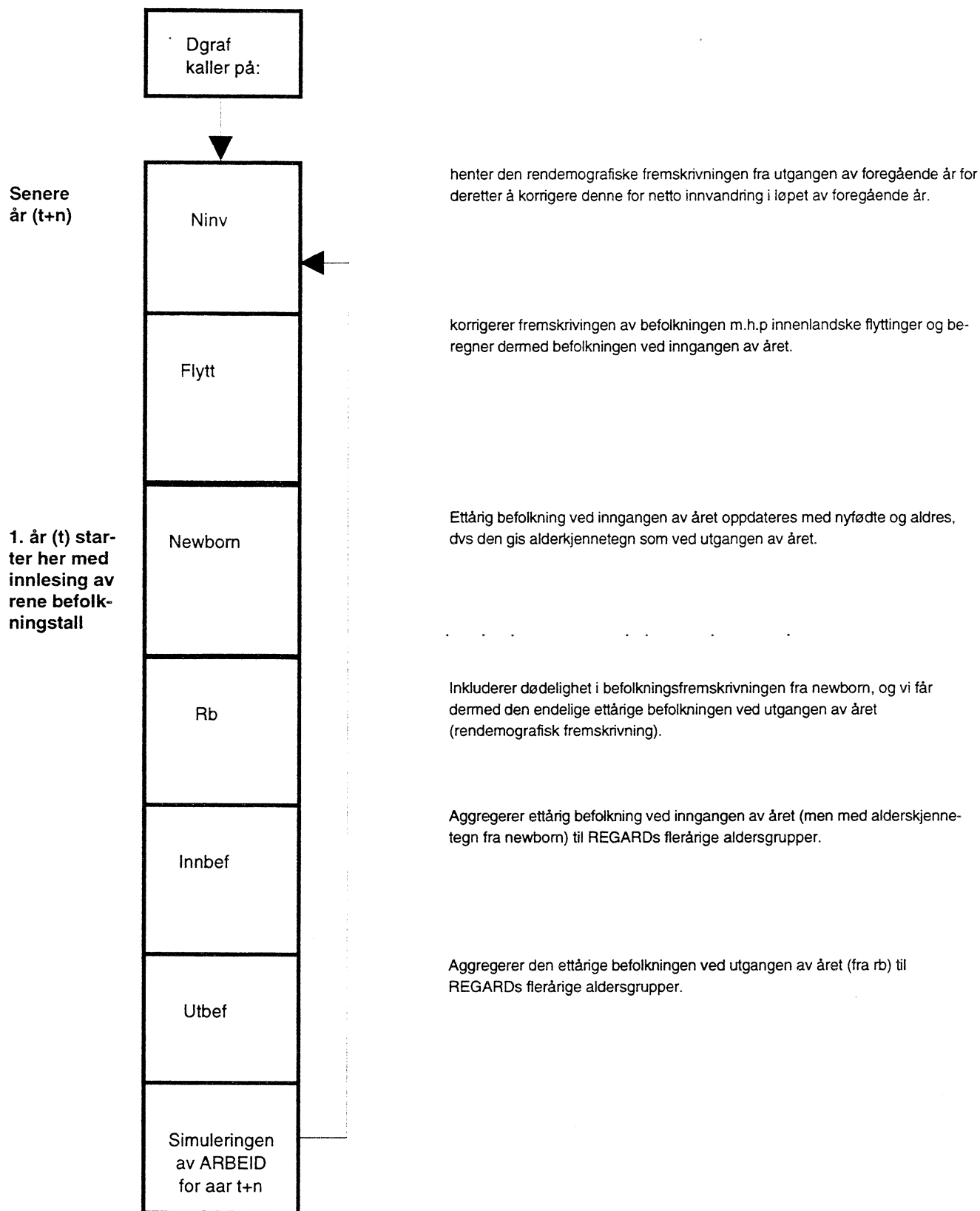
Skjematisk oversikt over formodellen



Oversikt over macroene som genererer likninger i ARBEID



Skjematisk oversikt over demografidelen i REGARD



**Utskrifter av macroer i
REGARD**

Styrings- og nyttemacroer i
REGARD


```
/* regard.src - Styringsmacro for simuleringer av regard-modellen */
/* Oversatt til POTroll av Rune Johansen, sist oppdatert 18.9.97 */

addfun main;
procedure main()
begin;
  print("INNLESING AV PARAMETRE FRA TERMINAL");
  /* parametre leses inn fra terminal (navn paa kjoring bestemmer */
  /* bl.a. navnet paa de databasene hvor resultatene lagres) */
  get navn "Navn paa kjoring:";
  get number aar1 "Forste simuleringsaar:";
  get number aar2 "Siste simuleringsaar:";

  navn = lower(navn);

  print("LAGER HJELPEMATRISENE CC,CU og CBARN");
  >>access ut0 type disk id /ssb/frisch/fame/regard/regard mode rw;
  >>search data ut0 w;
  &laghjelpmat;
  >>delsearch all;
  >>delaccess all;

  print("KOPIERING AV GRUNNLAGSDATA TIL ARBEIDSDATABASER");
  /* kopierer grunnlagsdata til aktuelle fame- og Troll databaser */
  &kopi; >>&(navn)

  print("SOK PAA DATABASER");

  /* aksesserer ut-databasene (arbeidsdatabasene) */
  >>access ut1 type fame id /ssb/frisch/fame/regard/data/&(navn).db
  >>mode rw;
  >>access ut2 type trolldb id /ssb/frisch/fame/regard/data/&(navn).trl
  >>mode rw;

  /* soker paa FAME databasen med MODAG og/eller KVARTS tall */
  >>access inn1 type fame id /ssb/frisch/fame/regard/grlag/formod.db mode r;
  >>search data inn1;

  /* aksesserer omraadet hvor hjelpevariable lagres */
  /* Her ligger ogsaa andelene for oppsplitting av MODAG- og KVARTS-tall */
  /* til fylkes- og regionsnivaa. Disse andelene er beregnet fra FNR-tall */
  >>access ut0 type disk id /ssb/frisch/fame/regard/regard mode rw;
  >>search data ut0 w;

  print("FORMODELLEN");
  /* Formodellen. Foretar eksogen tilrettelegging, aggregering fra */
  /* fylkesvist nasjonalregnskap, og aggregering fra fylke til */
  /* regioner */
  &formod; >>&(aar1)

  /* yrkesprosenter */
  /* eksogen beregning av yrkesprosenter ved hjelp av ny versjon */
  /* av macroen yprmod. Endret 25.09.96 RJO */
  &yprmod2; >>&(aar1) &(aar2)

  >>delsearch all;
  >>delaccess all;

  /* forste og siste simuleringsaar bestemmer skrankene i for-lokka */
  fra = aar1;
  til = aar2;

  print("HER STARTER SIMULERINGEN");
  /* simulerer for ett aar av gangen, fordi man maa oppdatere */
  /* demografiske data hvert aar, vi har derfor en vekselvirkning */
  /* mellom demografidel og kjoring av modellen */
  for(i=fra;i<=til;i=i+1)
  begin;
```

```
/* simuleringsaaret (som tekststreng) */
aar = convert(i)||"A";

print("SOK PAA DATABASER");

/* soker paa resultatbasene. Bruker modus 'u' (oppdatering) */
>>access ut1 type fame id /ssb/frisch/fame/regard/data/&(navn).db
>>mode u;
>>access ut2 type trollldb id /ssb/frisch/fame/regard/data/&(navn).trl
>>mode u;

/* aksesserer omraadet hvor hjelpevariable lagres */
>>access ut0 type disk id /ssb/frisch/fame/regard/regard mode r;
>>search data ut0;

/* aksesserer omraade hvor historiske befolkningsdata lagres */
>>access ut4 type trollldb id /ssb/frisch/fame/regard/grlag/historie.trl
>>mode r;

print("DEMOGRAFIDELLEN");
/* kaller paa REGARDS demografidel, dvs macroer som henter */
/* inn befolkningen ved utgangen av foregaaende aar, konverterer */
/* fra fleraarige til ettaarige aldersgrupper ved hjelp av */
/* gitte fordelingsmatriser, og foretar saa oppdatering av */
/* befolkningen med nyfodte, dode, netto innvandring og flytting */
&dgraf;
>>&(i)
>>&(fra)

>>delsearch all;
>>delaccess all;
>>delsave all;

print("SOK PAA DATABASER");
>>access ut1 type fame id /ssb/frisch/fame/regard/data/&(navn).db
>>mode u;
>>access ut2 type trollldb id /ssb/frisch/fame/regard/data/&(navn).trl
>>mode r;
>>search data ut1 w;
>>search data ut2;

/* aksesserer omraadet hvor hjelpevariable lagres */
>>access ut0 type disk id /ssb/frisch/fame/regard/regard mode r;
>>search data ut0;

print("SELVE SIMULERINGEN");
>>usemod arbeid;
>>simulate;
>>simstart &(aar);
>>dosim 1;
>>filesim ut1;

>>delsearch all;
>>delaccess all;

end;

print("SIMULERINGEN ER AVSLUTTET");

print("SOK PAA DATABASER");
>>access ut1 type fame id /ssb/frisch/fame/regard/data/&(navn).db
>>mode u;
>>access ut2 type trollldb id /ssb/frisch/fame/regard/data/&(navn).trl
>>mode u;

/* aksesserer omraade hvor historiske befolkningsdata lagres */
>>access ut4 type trollldb id /ssb/frisch/fame/regard/grlag/historie.trl
>>mode r;
```

```
/* aksesserer omraadet hvor hjelpevariable lagres */
>>access ut0 type disk id /ssb/frisch/fame/regard/regard mode r;
>>search data ut0;

print("NETTO INNVANDRING OG FLYTTING BEREGNES");
&ninv; >>&(til)
&flytt; >>&(til)

>>delsearch all;
>>delaccess all;

print("SOK PAA DATABASES FOR AA LAGE RESULTATMATRISER PAA TEKST-FORMAT");
>>access ut1 type fame id /ssb/frisch/fame/regard/data/&(navn).db
>>mode r;
>>access ut2 type trolldb id /ssb/frisch/fame/regard/data/&(navn).trl
>>mode r;

>>access ut type disk id /ssb/frisch/fame/regard/res mode rw;
>>search data ut w;

>>access ut0 type disk id /ssb/frisch/fame/regard/regard mode r;
>>search data ut0;

print("RESULTATENE EKSPORTERES");
/* Lager tekst-matriser, som inneholder resultattall for befolkning, */
/* flytting, arbeidsstyrke, sysselsetting og markedsleie */
&export; >>&(navn) &(aar1) &(aar2)

>>delsearch all;
>>delaccess all;

end;
```

```
/* kopi.src - kopierer eksogene historiske data til de databasene */
/* REGARD opererer paa (arbeidsdatabasene). Tidsseriene blir liggende */
/* paa en FAME database, resten paa en TROLL-database (.trl) */
/* Programmert av Rune Johansen. Sist oppdatert 7.11.1996 */

addfun main;
procedure main()
begin;

get navn"navn paa kjoring/omraade:";

navn = lower(navn);

/* aksesserer TROLL-database med grunnlagstall */
>>access inn type trolldb id /ssb/frisch/fame/regard/grlag/grlag.trl
>>mode r;
>>search data inn;
/* arbeidsdatabase som inneholder matriser og vektorer (bl.a. bef-tall). */
>>access ut2 type trolldb id /ssb/frisch/fame/regard/data/&(navn).trl
>>mode rw;

for(a=1991;a<=2015;a=a+1)
begin;
  if (a<=2001) then
  begin;
    /* befolkningstall bare estimert frem til 2002 */
    >>dofile ut2_bef&(a)=bef&(a);
    >>dofile ut2_nfb&(a)=nfb&(a);
    for(j=1;j<=3;j=j+1)
    begin;
      >>dofile ut2_bef&(j)&(a)=bef&(j)&(a);
      >>dofile ut2_nfb&(j)&(a)=nfb&(j)&(a);
    end;
  end;
  /* overlevelsesrater er estimert for alle aarene fra 1990 til 2015 */
  >>dofile ut2_sk&(a) = sk&(a);
  >>dofile ut2_sm&(a) = sm&(a);
end;

for(j=1;j<=3;j=j+1)
begin;
  >>dofile ut2_bef&(j)1990=bef&(j)1990;
  >>dofile ut2_bef&(j)2002=bef&(j)2002;
end;

>>dofile ut2_fbk=fbk;
>>dofile ut2_fbm=fbm;
>>dofile ut2_flb1=flb1;
>>dofile ut2_flb2=flb2;
>>dofile ut2_flv1=flv1;
>>dofile ut2_flv2=flv2;
>>dofile ut2_flv3=flv3;
>>dofile ut2_fv1=fv1;
>>dofile ut2_fv2=fv2;
>>dofile ut2_fv3=fv3;
>>dofile ut2_gu1=gu1;
>>dofile ut2_gu2=gu2;
>>dofile ut2_gu3=gu3;

>>delsearch all;
>>delaccess all;

/* aksess og sok paa grunnlagsdatabase med tidsserier */
>>access inn1 type fame id /ssb/frisch/fame/regard/grlag/grlag.db
>>mode r;
>>search data inn1;
/* tidsserier for variable i hovedmodellen ARBEID lagres paa en */
/* FAME database */
>>access ut1 type fame id /ssb/frisch/fame/regard/data/&(navn).db
```

```

>>mode rw;

for (k=1;k<=7;k=k+1)
begin;
  /* hovedlokken loper over region */
  >>dofile ut1_arbst&(k) = arbst&(k);
  >>dofile ut1_syss&(k) = syss&(k);
  >>dofile ut1_barn&(k) = barn&(k);
  >>dofile ut1_bef&(k) = bef&(k);
  >>dofile ut1_mgam&(k) = mgam&(k);
  for(i=1;i<=2;i=i+1)
  begin;
    for(l=3;l<=9;l=l+1)
    begin;
      >>dofile ut1_jb&(i)&(l)&(k) = jb&(i)&(l)&(k);
      for(j=1;j<=3;j=j+1)
      begin;
        >>dofile ut1_as&(i)&(l)&(j)&(k) = as&(i)&(l)&(j)&(k);
        >>dofile ut1_s.as&(i)&(l)&(j)&(k) = s.as&(i)&(l)&(j)&(k);
        >>dofile ut1_s.in&(i)&(l)&(j)&(k) = s.in&(i)&(l)&(j)&(k);
        >>dofile ut1_s.ut&(i)&(l)&(j)&(k) = s.ut&(i)&(l)&(j)&(k);
        >>dofile ut1_in&(i)&(l)&(j)&(k) = in&(i)&(l)&(j)&(k);
        >>dofile ut1_ut&(i)&(l)&(j)&(k) = ut&(i)&(l)&(j)&(k);
        >>dofile ut1_butg&(i)&(l)&(j)&(k) = butg&(i)&(l)&(j)&(k);
        >>dofile ut1_b&(i)&(l)&(j)&(k) = b&(i)&(l)&(j)&(k);
      end;
      >>dofile ut1_b&(i)&(l)&(k) =
      >>b&(i)&(l)1&(k)+b&(i)&(l)2&(k)+b&(i)&(l)3&(k);
    end;
    for(l=1;l<=2;l=l+1)
    begin;
      >>dofile ut1_barn&(i)&(l)&(k) = barn&(i)&(l)&(k);
      >>dofile ut1_bgm&(i)&(l)&(k) = bgm&(i)&(l)&(k);
      >>dofile ut1_inb&(i)&(l)&(k) = inb&(i)&(l)&(k);
      >>dofile ut1_s.inb&(i)&(l)&(k) = s.inb&(i)&(l)&(k);
      >>dofile ut1_utb&(i)&(l)&(k) = utb&(i)&(l)&(k);
      >>dofile ut1_s.utb&(i)&(l)&(k) = s.utb&(i)&(l)&(k);
      for(k2=1;k2<=7;k2=k2+1)
      begin;
        if (k2 <> k) then
        begin;
          >>dofile ut1_flrb&(i)&(l)&(k)&(k2)=flrb&(i)&(l)&(k)&(k2);
        end;
      end;
    end;
    for(l=1;l<=9;l=l+1)
    begin;
      >>dofile ut1_butg&(i)&(l)&(k) = butg&(i)&(l)&(k);
      >>dofile ut1_inv&(i)&(l)&(k) = inv&(i)&(l)&(k);
      >>dofile ut1_utv&(i)&(l)&(k) = utv&(i)&(l)&(k);
    end;
    for(j=1;j<=3;j=j+1)
    begin;
      >>dofile ut1_l&(i)&(j)&(k) = l&(i)&(j)&(k);
      >>dofile ut1_gamma&(i)&(j)&(k) = gamma&(i)&(j)&(k);
      >>dofile ut1_xml&(i)&(j)&(k) = xml&(i)&(j)&(k);
      >>dofile ut1_np&(i)&(j)&(k) = np&(i)&(j)&(k);
      >>dofile ut1_s.np&(i)&(j)&(k) = s.np&(i)&(j)&(k);
    end;
  end;
end;

end;

for(i=1;i<=2;i=i+1)
begin;
  for(j=1;j<=3;j=j+1)
  begin;
    >>dofile ut1_l&(i)&(j)8 = l&(i)&(j)8;
    >>dofile ut1_gamma&(i)&(j)8 = gamma&(i)&(j)8;
  end;
end;

```

```
>>dofile ut1_jnp&(i)&(j) = jnp&(i)&(j);
for(l=3;l<=9;l=l+1)
begin;
  if (l==3) then
  begin;
    if (j<>3) then
    begin;
      >>dofile ut1_m&(i)&(l)&(j) = m&(i)&(l)&(j);
    end;
  end; else
  begin;
    >>dofile ut1_m&(i)&(l)&(j) = m&(i)&(l)&(j);
  end;
  >>dofile ut1_yp&(i)&(l)&(j) = yp&(i)&(l)&(j);
  >>dofile ut1_y2m&(i)&(l)&(j) = y2m&(i)&(l)&(j);
end;
end;
end;

>>dofile ut1_nvpl16 = nvpl16;
>>dofile ut1_ls60u = ls60u;
>>dofile ut1_mgam8 = mgam8;
>>dofile ut1_bruttoin = bruttoin;
>>dofile ut1_nettoin = nettoin;
>>dofile ut1_jas = jas;
>>dofile ut1_bm133 = bm133;
>>dofile ut1_bm233 = bm233;

>>delsearch all;
>>delaccess all;

end;
```

```
/* laggrlag.src */
/* Macro som henter inn de MODAG eller KVARTS tall som skal brukes i REGARD */
/* og lagrer de paa en database */
/* Maa kjores hver gang en tar i bruk en ny MODAG eller KVARTS-kjoring som */
/* grunnlag i REGARD. KVARTS-tallene maa gjores om til aarlige tall */
/* Programmert av Rune Johansen. Siste versjon 18.9.1997 */

addfun main;
procedure main()
begin;

  get s"Skal tallene hentes fra MODAG eller KVARTS (M/K):";

  /* henter lista med sektorkoder */
  >>access d0 type disk id ./ mode r;
  >>search data d0;
  liste = getdata("modag");
  nls = nvals(liste);
  >>delsearch all;
  >>delaccess all;

  /* aksesserer utdatabasen hvor MODAG/KVARTStall som skal brukes blir */
  /* lagret */
  >>access ut0 type fame id /ssb/frisch/fame/regard/grlag/formod.db mode rw;

  if (s == "M" or s == "m") then
  begin;
    /* MODAG tall */
    get up 2 navn1"Navn paa MODAG-databasen:";
    navn1 = lower(navn1);
    >>access in0 type fame id /ssb/frisch/fame/regard/grlag/&(navn1).db
    >>mode r;
    >>search data in0;

    for(i=1;i<=nls;i=i+1)
    begin;
      se = values(liste,i);
      >>dofile ut0_jks&(se) = jks&(se);
      >>dofile ut0_q&(se) = q&(se);
      >>dofile ut0_n&(se) = n&(se);
    end;

    >>dofile ut0_nb1619 = nb1619;
    >>dofile ut0_nvpl20 = nvpl20;
    >>dofile ut0_nb2024 = nb2024;
    >>dofile ut0_nt = nt;

  end; else
  begin;
    get up 2 navn1"Navn paa MODAG-databasen:";
    get up 2 navn2"Navn paa KVARTS-databasen:";
    navn1 = lower(navn1);
    navn2 = lower(navn2);
    /* KVARTS-tallene maa gjores om til aarlige storrelser ved hjelp */
    /* av TROLL-kommandoen compact */
    >>access in0 type fame id /ssb/frisch/fame/regard/grlag/&(navn1).db
    >>mode r;
    >>access in1 type fame id /ssb/frisch/fame/regard/grlag/&(navn2).db
    >>mode r;

    for(i=1;i<=nls;i=i+1)
    begin;
      se = values(liste,i);
      >>dofile ut0_jks&(se) = compact(in1_jks&(se),-1,1);
      >>dofile ut0_q&(se) = compact(in1_q&(se),-1,1);
      >>dofile ut0_nw&(se) = compact(in1_nw&(se),0,1);
    end;

    >>dofile ut0_nb1619 = compact(in1_nb1619,0,1);
```

```
>>dofile ut0_nvpl20 = compact(in1_nvpl20,0,1);
>>dofile ut0_nb2024 = compact(in1_nb2024,0,1);
>>dofile ut0_nt = compact(in1_nt,0,1);
>>dofile ut0_ns = compact(in1_ns,0,1);

>>do ns92s = 0;
>>do ns93s = 0;
>>do ns94s = 0;
>>do ns95s = 0;
>>do ns93k = 0;
>>do ns94k = 0;
>>do ns95k = 0;
>>do nsmod = 0
for(i=1;i<=nls;i=i+1)
begin;
    se = values(liste,i);
    >>+in0_ns&(se)
end;
>>;

for(i=1;i<=nls;i=i+1)
begin;
    se = values(liste,i);
    /* Mangler sysselsetting etter sektor for selvstendig naerings- */
    /* drivende i KVARTS. Splitter opp etter samme fordeling som den */
    /* siste MODAG-kjoringen. */
    >>dofile ut0_ns&(se) = (ut0_ns/nsmod)*in0_ns&(se);
    >>dofile ut0_n&(se) = ut0_ns&(se) + ut0_nw&(se);
end;

end;

>>delsearch all;
>>delaccess all;

end;
```



```
/* medit.src - Hjelpemacro som leser inn ett og ett tall, og bygger opp */  
/* en matrise av spesifisert dimensjon */  
/* Brukes i REGARD til innlesing av tall fra fylkesvist nasjonalregnskap. */  
/* Programmert av Rune Johansen. Siste versjon 5 juni 1996 */
```

```
addfun main;  
procedure main()  
begin;  
  
  get navn; /* navn paa matrise (brprod,invest eller syss) */  
  get number rad; /* antall rader */  
  get number kol; /* antall kolonner */  
  
  >>do y=combine(  
    for(i=1;i<=rad;i=i+1)  
    begin;  
      for(j=1;j<=kol;j=j+1)  
      begin;  
        get number x;  
        if (i==1 and j==1) then  
          >>&(x)  
        else  
          >>,&(x)  
      end;  
    end;  
  >>);  
  
  >>dofile &(navn) = transp(reshape(y,&(kol),&(rad)));  
  
end;
```

```
/* laghjelpmat.src - Macro som lager hjelpematriser til oppsplitting */  
/* fra REGARD's fleraarige til ettaarig aldersinndeling */  
/* Produserer de tre matrisene CC, CU og CBARN */  
/* Programmert av Rune Johansen. Sist oppdatert 5.2.1997 */
```

```
addfun main;  
procedure main()  
begin;  
  
    &aggcc;  
  
    >>do nuller = crmat(11,86,0);  
    >>do cc = partcomb(2,2,cc,nuller,nuller,cc);  
    >>dofile cc = transp(cc);  
  
    &aggcu;  
  
    >>do nuller = crmat(7,59,0);  
    >>do cu = partcomb(2,2,cu,nuller,nuller,cu);  
    >>dofile cu = transp(cu);  
  
    &aggcbarn;  
  
    >>dofile cbarn = transp(cbarn);  
  
end;
```

```
/* aggcc.src - macro som produserer (aggregerings)matrisen cc */
/* bestaaende av enere og nuller, og som angir overgangen mellom */
/* ettaarige og fleraarige aldersgruppeinndelinger i REGARD. */
/* Kaller paa matrisen lagmat2, som lager selve matrisen ved hjelp */
/* av den gitte spesifikasjonen. */
/* Programmert av Rune Johansen. Sist oppdatert 24. juli 1996. */
```

```
addfun main;
procedure main()
begin;
&lagmat2;
>>cc 86 11
>> 1 = 1 + 2 + 3 + 4 + 5 + 6 + 7 *
>> 2 = 8 + 9 + 10 + 11 + 12 + 13 + 14 + 15 + 16 *
>> 3 = 17 + 18 + 19 + 20 *
>> 4 = 21 + 22 + 23 + 24 + 25 *
>> 5 = 26 + 27 + 28 + 29 + 30 + 31 + 32 + 33 + 34 + 35 *
>> 6 = 36 + 37 + 38 + 39 + 40 + 41 + 42 + 43 + 44 + 45 *
>> 7 = 46 + 47 + 48 + 49 + 50 + 51 + 52 + 53 + 54 + 55 + 56 + 57 + 58
>>   + 59 + 60 *
>> 8 = 61 + 62 + 63 + 64 + 65 + 66 + 67 *
>> 9 = 68 + 69 + 70 + 71 + 72 + 73 + 74 + 75 *
>>10 = 76 + 77 + 78 + 79 + 80 + 81 + 82 + 83 + 84 + 85 *
>>11 = 86 *
>>**
end;
```

```
/* aggcu.src - macro som produserer (aggregerings)matrisen cu */
/* bestaaende av enere og nuller, og som angir overgangen mellom */
/* ettaarige og fleraarige aldersgruppeinndelinger i REGARD, for */
/* de yrkesaktive, dvs 7 fleraarige aldersgrupper (3-9) som tilsvarer */
/* 59 ettaarige aldersgrupper (16-74) */
/* Kaller paa matrisen lagmat2, som lager selve matrisen ved hjelp */
/* av den gitte spesifikasjonen. */
/* Programmert av Rune Johansen. Sist oppdatert 24. juli 1996. */
```

```
addfun main;
procedure main()
begin;
&lagmat2;
>>cu 59 7
>> 1 = 1 + 2 + 3 + 4 *
>> 2 = 5 + 6 + 7 + 8 + 9 *
>> 3 = 10 + 11 + 12 + 13 + 14 + 15 + 16 + 17 + 18 + 19 *
>> 4 = 20 + 21 + 22 + 23 + 24 + 25 + 26 + 27 + 28 + 29 *
>> 5 = 30 + 31 + 32 + 33 + 34 + 35 + 36 + 37 + 38 + 39 + 40 + 41 + 42
>>   + 43 + 44 *
>> 6 = 45 + 46 + 47 + 48 + 49 + 50 + 51 *
>> 7 = 52 + 53 + 54 + 55 + 56 + 57 + 58 + 59 *
>>**
end;
```

```
/* aggcbarn.src - Lager hjelpematriksen cbarn, som er en matrise */  
/* av enere og nuller, og som angir overgangen fra fleraarige til */  
/* ettaarig alder for barn, dvs 2 fleraarige aldersgrupper (1-2), */  
/* tilsvarende 16 ettaarige grupper (0-15) */  
/* Matrisen blir laget ved hjelp av et kall på hjelpemacroen lagmat2. */  
/* Programmert av Rune Johansen. Sist oppdatert 24.juli 1996 */
```

```
addfun main;  
procedure main()  
begin;  
&lagmat2;  
>>cbarn 16 2  
>> 1 = 1 + 2 + 3 + 4 + 5 + 6 + 7 *  
>> 2 = 8 + 9 + 10 + 11 + 12 + 13 + 14 + 15 + 16 *  
>>**  
end;
```

```
/* lagmat2 - Macro som konverterer brukerens aggregeringsspesifikasjon, */  
/* og som produserer aggregeringsmatrisene. Dette er matriser */  
/* som bestaar av 1'ere og nuller, og selve aggregeringen utfoeres */  
/* ved aa multiplisere disse med de aktuelle datamatrisene. */  
/* Spesialversjon for aa lage hjelpmatriser i demografidelen i REGARD */  
/* Programmert av Rune Johansen. Siste versjon 18 juli 1996 */
```

```
addfun main;  
procedure main()  
begin;  
  
  i=1;i1=1;j=1;  
  
  get navn;  
  get number n1;  
  get number n2;  
  
  matrise = crmat(n2,n1,0);  
  
  GET y;  
  while (y <> "**")  
  begin;  
    /* finner i hvilken rad neste 1-tall skal plasseres*/  
    i = convert(y,"numeric");  
    GET y; /* likhetstegn,pluss eller stjerne */  
    while (y <> "**")  
    begin;  
      GET y;  
      /* finner hvilken kolonne 1-tallet skal plasseres */  
      j = convert(y,"numeric");  
      /* plasserer et 1-tall i rad nr i og kolonne nr. j */  
      matrise=setrep(matrise,1,i,j);  
      GET y;  
    end;  
    GET y;  
  end;  
  
  /* den konstruerte matrisen lagres paa fil */  
  PUTDATA(matrise,navn,-1);  
  
end;
```

```
/* colsplit - erstatter den gamle colsplit-kommandoen i Troll */  
/* Programmert av Rune Johansen. Sist oppdatert 1.8.96 */
```

```
addfun main;  
procedure main()  
begin;  
  get navn"Navn paa matrisen som skal splittes";  
  
  >>do x = &(navn);  
  >>do nc = noc.(x);  
  nkol = getdata("NC",1);  
  for(i=1;i<=nkol;i=i+1)  
  begin;  
    >>dofile &(navn)c&(i)=submat(x,,col(x)==&(i));  
  end;  
end
```

```

/* kombiner.src - macro som kan erstatte den gamle PARTCOMB-kommandoen */
/* i Troll. Brukes ved versjoner lavere enn 1.0 */
/* Programmert av Rune Johansen. Sist oppdatert 1.8.96 */

addfun main;
procedure main()
begin;

  get navn"Navn paa resultatmatrisen";
  get number m"Antall matriser i hoyden";
  get number n"antall matriser i bredden";
  for(i=1;i<=m;i=i+1)
  begin;
    for(j=1;j<=n;j=j+1)
    begin;
      get mat;
      >>dofile matrise&(i)&(j)=&(mat);
    end;
  end;

  >>do nc=noc.(matrisel1);
  ncol=getdata("NC",1);

  for(i=1;i<=n;i=i+1)
  begin;
    >>do nc=noc.(matrisel&(i));
    ncol=getdata("NC",1);
    for(j=1;j<=m;j=j+1)
    begin;
      >>do x=matrise&(j)&(i);
      for(k=1;k<=ncol;k=k+1)
      begin;
        >>dofile kol&(i)&(j)&(k)=submat(x,,col(x)==&(k));
      end;
    end;

    for(k=1;k<=ncol;k=k+1)
    begin;
      >>dofile soyle&(i)&(k)=combine(
      for(j=1;j<=m;j=j+1)
      begin;
        if (j<m) then
          >>kol&(i)&(j)&(k),
        else
          >>kol&(i)&(j)&(k)
        end;
      >>);
    end;
  end;

  >>dofile nymat = soylel1;
  >>do nc=noc.(matrisel1);
  ncol=getdata("NC",1);
  nn=ncol;
  for(k=2;k<=ncol;k=k+1)
  begin;
    >>dofile nymat=addcol(nymat,&(k)-1,soylel&(k));
  end;
  for(j=2;j<=n;j=j+1)
  begin;
    >>do nc=noc.(matrisel&(j));
    ncol=getdata("NC",1);
    for(k=1;k<=ncol;k=k+1)
    begin;
      m=nn+k-1;
      >>dofile nymat=addcol(nymat,&(m),soyle&(j)&(k));
    end;
    nn=nn+ncol;
  end;

```



```
end;  
>>dofile &(navn)=nymat;
```

```
end;
```

Formodellen

```
/* formod.src - macro for aa lese inn eksogene data fra Modag og Mosart */
/* og fordele disse paa regioner. */
/* Programmert av Rune Johansen. Sist oppdatert 3.10.1996 */

addfun main;
procedure main()
begin;

    /* Basisaar. Trengs blant annet til beregning av total arbeidsstyrke */
    get number aar;

    /* sektorliste */
    sektor = getdata("modag");
    ns = nvals(sektor);

    print("HENTER FNR-TALL, OG BEREGNER ANDELER");
    /* Beregning av fordelingskoeffisienter for bruttoprodukt, */
    /* bruttoinvestering og sysselsetning, paa bakgrunn av */
    /* fylkesvist nasjonalregnskap. Aggregerer først tallene til */
    /* Modag-nivaa */
    &fylkvis;

    print("HENTER MODAG-TALL");
    /* Henter inn tall fra Modag og fordeler paa regioner */

    /* bruttoinvesteringer etter sektor (JS) */
    &invest2;

    /* qs-likninger (bruttoprodukt) */
    &lagqseq;

    /* Sysselsetningstall */
    &syss;

    /* Arbeidsstyrke */
    &arbstyrk; >>&(aar)

    print("FRA FYLKE TIL REGION");
    /* aggregerer fra fylke til sju regioner */

    /* bruttoprodukt */
    for(i=1;i<=ns;i=i+1)
    begin;
        se = values(sektor,i);
        >>dofile ut1_qs&(se)1=ut1_qs&(se)02+ut1_qs&(se)03;
        >>dofile ut1_qs&(se)2=ut1_qs&(se)01+ut1_qs&(se)06+ut1_qs&(se)07+
        >>ut1_qs&(se)08;
        >>dofile ut1_qs&(se)3=ut1_qs&(se)04+ut1_qs&(se)05;
        >>dofile ut1_qs&(se)4=ut1_qs&(se)09+ut1_qs&(se)10+ut1_qs&(se)11;
        >>dofile ut1_qs&(se)5=ut1_qs&(se)12+ut1_qs&(se)14+ut1_qs&(se)15;
        >>dofile ut1_qs&(se)6=ut1_qs&(se)16+ut1_qs&(se)17;
        >>dofile ut1_qs&(se)7=ut1_qs&(se)18+ut1_qs&(se)19+ut1_qs&(se)20;
        >>dofile ut1_qs&(se)8=ut1_qs&(se)99;
    end;

    /* bruttoinvestering */
    for(i=1;i<=ns;i=i+1)
    begin;
        se = values(sektor,i);
        >>dofile ut1_js&(se)1=ut1_js&(se)02+ut1_js&(se)03;
        >>dofile ut1_js&(se)2=ut1_js&(se)01+ut1_js&(se)06+ut1_js&(se)07+
        >>ut1_js&(se)08;
        >>dofile ut1_js&(se)3=ut1_js&(se)04+ut1_js&(se)05;
        >>dofile ut1_js&(se)4=ut1_js&(se)09+ut1_js&(se)10+ut1_js&(se)11;
        >>dofile ut1_js&(se)5=ut1_js&(se)12+ut1_js&(se)14+ut1_js&(se)15;
        >>dofile ut1_js&(se)6=ut1_js&(se)16+ut1_js&(se)17;
        >>dofile ut1_js&(se)7=ut1_js&(se)18+ut1_js&(se)19+ut1_js&(se)20;
        >>dofile ut1_js&(se)8=ut1_js&(se)99;
```

```
end;
```

```
/* sysselsetning */
```

```
for(i=1;i<=ns;i=i+1)
```

```
begin;
```

```
  se = values(sektor,i);
```

```
  >>dofile ut1_ls&(se)1=ut1_ls&(se)02+ut1_ls&(se)03;
```

```
  >>dofile ut1_ls&(se)2=ut1_ls&(se)01+ut1_ls&(se)06+ut1_ls&(se)07+
```

```
  >>ut1_ls&(se)08;
```

```
  >>dofile ut1_ls&(se)3=ut1_ls&(se)04+ut1_ls&(se)05;
```

```
  >>dofile ut1_ls&(se)4=ut1_ls&(se)09+ut1_ls&(se)10+ut1_ls&(se)11;
```

```
  >>dofile ut1_ls&(se)5=ut1_ls&(se)12+ut1_ls&(se)14+ut1_ls&(se)15;
```

```
  >>dofile ut1_ls&(se)6=ut1_ls&(se)16+ut1_ls&(se)17;
```

```
  >>dofile ut1_ls&(se)7=ut1_ls&(se)18+ut1_ls&(se)19+ut1_ls&(se)20;
```

```
  >>dofile ut1_ls&(se)8=ut1_ls&(se)99;
```

```
end;
```

```
end;
```

```

/* fylkvis.src - Beregner fordelingskoeffisienter ved hjelp av */
/* fylkesvist nasjonalregnskap. Aggregerer først fra NR-nivaa til Modag */
/* Programmert av Rune Johansen. Sist oppdatert 10.07.96 */

addfun main;
procedure main()
begin;

  /* henter vektorer som inneholder koder for fylke og sektor */
  fylke = getdata("fylkkode");
  nf = nvals(fylke);
  sektor = getdata("modag");
  ns = nvals(sektor);

  /* lager aggregeringsmatrisene */
  &afratilps; /* bruttoprodukt */
  &afratiljs; /* investering */
  &afratilsy; /* sysselsetting */

  /* Foer vi kan aggregere maa vi justere NR-sektor 23611 */
  /* (utenriks sjofart) i matrisen syss(elsetting), slik at */
  /* utlendingene ikke telles med. Dette gjøres ved aa gange */
  /* alle elementer i den korresponderende raden av FNR-matrisen */
  /* syss med en faktor fac (vi antar at andelen utlendinger i */
  /* utenriks sjofart er konstant i hvert fylke) */
  lsyss=getdata("syss");
  nrad=nor.(lsyss);
  liste=getdata("lfrasy");
  fac=1-(28490/43940);
  >>do x=combine(
  for(i=1;i<=nrad;i=i+1)
  begin;
    nrkode=values(liste,i);
    if (nrkode=="23611") then
      >>&(fac),
    else begin;
      if (i<nrad) then
        >>1,
      else
        >>1
    end;
  end;
  >>);
  >>do y=x+transp(seq(20)*0);
  >>dofile syss=y*syss; /* elementvis multiplikasjon */

  /* aggregerer fra NR-nivaa til Modag */
  >>dofile bruttoprod = matmult(aggps,brprod);
  >>dofile investering = matmult(aggjs,invest);
  >>dofile syssel = matmult(aggjs,syss);

  /* bruttoprodukt, investering og sysselsetning er naa paa Modag-nivaa, */
  /* og fylkesvist fordelt. Regner saa ut koeffisientene. */

  >>do x = seq(&(nf))*0+1;

  /* Brutttoprodukt */
  >>do nevner = matmult(brutttoprod,x);
  /* ekspanderer nevneren slik at den faar samme dimensjon som matrisen */
  >>do nevner = nevner + transp(seq(&(nf))*0);
  >>dofile fqs = bruttoprod/ if nevner == 0 then 1 else nevner;

  /* Bruttoinvesteringer */
  >>do nevner = matmult(investering,x);
  /* ekspanderer nevneren slik at den faar samme dimensjon som matrisen */
  >>do nevner = nevner + transp(seq(&(nf))*0);
  >>dofile fjs = investering/ if nevner == 0 then 1 else nevner;

  /* Sysselsetning */

```

```
>>do nevner = matmult(syssel,x);  
/* ekspanderer nevneren slik at den faar samme dimensjon som matrisen */  
>>do nevner = nevner + transp(seq(&(nf))*0);  
>>dofile fls = syssel/ if nevner == 0 then 1 else nevner;
```

```
end;
```

```
/* afratiljs.src - macro som lager aggregeringsmatrisen for investering */
/* Kaller paa macroen lagmat. */
/* Programmert av Rune Johansen. Sist oppdatert 26. juli 1996 */

addfun main;
procedure main()
begin;
&lagmat;
>>js
>>11 = 83010 *
>>12 = 83020 *
>>13 = 83051 *
>>14 = 83052 *
>>15 = 83151 + 83152 + 83153 + 83154 + 83155 + 83156 + 83157 + 83158 +
>>      83159 + 83160 + 83170 + 83180 + 83190 *
>>25 = 83100 + 83130 + 83140 + 83201 + 83202 + 83203 + 83204 + 83221 +
>>      83222 + 83243 + 83244 + 83245 + 83246 + 83250 + 83265 +
>>      83261 + 83262 + 83266 + 83361 + 83362 + 83363 *
>>34 = 83211 + 83212 + 83213 *
>>37 = 83241 + 83242 + 83247 *
>>40 = 83231 + 83232 *
>>43 = 83271 + 83273 + 83274 + 83275 *
>>45 = 83281 + 83286 + 83287 + 83291 + 83293 + 83296 + 83297 + 83300 +
>>      83311 + 83313 + 83314 + 83321 + 83323 + 83331 + 83334 + 83340 +
>>      83353 + 83354 + 83355 *
>>50 = 83351 + 83352 *
>>71 = 83401 + 83402 + 83403 *
>>55 = 83451 + 83452 + 83453 + 83454 *
>>81 = 83501 + 83505 + 83510 + 83521 *
>>64 = 83111 + 83608 *
>>65 = 83112 + 83611 *
>>74 = 83601 + 83603 + 83605 + 83604 + 83602 + 83613 + 83620 + 83631 +
>>      83632 + 83633 + 83641 + 83642 *
>>63 = 83651 + 83652 + 83655 + 83661 + 83662 + 83663 *
>>83 = 82704 *
>>85 = 83405 + 83551 + 83553 + 83502 + 83527 + 83700 + 83711 + 83713 +
>>      83720 + 83730 + 83741 + 83742 + 83744 + 83745 + 83747 + 83748 +
>>      83800 + 83852 + 83851 + 83853 + 83859 + 83900 + 83910 + 83921 +
>>      83922 + 83926 + 83927 + 83930 + 86800 + 86851 + 86853 + 86854 +
>>      86910 + 86921 + 86926 *
>>"92S" = 84752 *
>>"93S" = 84800 *
>>"94S" = 84852 + 84851 *
>>"95S" = 84601 + 84631 + 84632 + 84730 + 84742 + 84745 + 84751 + 84921 *
>>"93K" = 85800 *
>>"94K" = 85851 + 85853 + 85854 *
>>"95K" = 85410 + 85751 + 85900 + 85921 *
>>**

end;
```

```
/* afratilps.src - macro som leser inn aggregeringsspesifikasjonen fra */
/* NR til MODAG for bruttoprodukt, og lager aggregeringsmatrisen. */
/* Kaller paa macroen lagmat. */
/* programmert av Rune Johansen. Sist oppdatert 26. juli 1996 */

addfun main;
procedure main()
begin;
&lagmat;
>>ps
>>11 = 22010 + 22015 + 23010 + 23014 *
>>12 = 23020 *
>>13 = 22051 + 23051 *
>>14 = 23052 *
>>15 = 23151 + 23152 + 23153 + 23154 + 23155 + 23156 + 23157 + 23158 +
>> 23159 + 23160 + 23170 + 23180 + 23190 *
>>25 = 23100 + 23130 + 23140 + 23201 + 23202 + 23203 + 23204 +
>> 23221 + 23222 + 23243 + 23244 + 23245 + 23246 + 23250 + 23265 +
>> 23261 + 23262 + 23266 + 23361 + 23362 + 23363 + 23372 *
>>34 = 23211 + 23212 + 23213 *
>>37 = 23241 + 23242 + 23247 *
>>40 = 23231 + 23232 *
>>43 = 23271 + 23273 + 23274 + 23275 *
>>45 = 23281 + 23286 + 23287 + 23291 + 23293 + 23296 + 23297 + 23300 +
>> 23311 + 23313 + 23314 + 23321 + 23323 + 23331 + 23334 + 23340 +
>> 23353 + 23354 + 23355 *
>>50 = 23351 + 23352 *
>>71 = 23401 + 23402 + 23403 *
>>55 = 22452 + 23451 + 23452 + 23453 + 23454 *
>>81 = 23501 + 23505 + 23510 + 23521 *
>>64 = 23111 + 23608 *
>>65 = 23112 + 23611 *
>>74 = 23601 + 23603 + 23605 + 23604 + 23602 + 23613 + 23620 + 23631 +
>> 23632 + 23633 + 23641 + 23642 *
>>63 = 23651 + 23652 + 23655 + 23661 + 23662 + 23663 + 23670 *
>>83 = 22704 *
>>85 = 22950 + 23405 + 23551 + 23553 + 23502 + 23527 +
>> 23700 + 23711 + 23713 + 23720 + 23730 + 23741 + 23742 + 23744 + 23745 +
>> 23747 + 23748 + 23800 + 23852 + 23851 + 23853 + 23859 + 23900 +
>> 23910 + 23921 + 23922 + 23926 + 23927 + 23930 +
>> 26800 + 26851 + 26853 + 26854 + 26910 + 26921 + 26926 *
>>"92S" = 24752 *
>>"93S" = 24800 *
>>"94S" = 24852 + 24851 + 24853 *
>>"95S" = 24601 + 24631 + 24632 + 24670 + 24730 + 24742 + 24745 +
>> 24751 + 24921 *
>>"93K" = 25800 *
>>"94K" = 25851 + 25853 + 25854 *
>>"95K" = 25410 + 25453 + 25751 + 25900 + 25921 *
>>**

end;
```



```
/* afratilsy.src - macro som lager aggregeringsmatrisen for sysselsetting */
/* Kaller paa macroen lagmat. */
/* Programmert av Rune Johansen. Sist oppdatert 26. juli 1996 */

addfun main;
procedure main()
begin;
  &lagmat;
  >>sy
  >>11 = 23010 + 23014 *
  >>12 = 23020 *
  >>13 = 23051 *
  >>14 = 23052 *
  >>15 = 23151 + 23152 + 23153 + 23154 + 23155 + 23156 + 23157 + 23158 +
  >>    23159 + 23160 + 23170 + 23180 + 23190 *
  >>25 = 23100 + 23130 + 23140 + 23201 + 23202 + 23203 + 23204 +
  >>    23221 + 23222 + 23243 + 23244 + 23245 + 23246 + 23250 + 23265 +
  >>    23261 + 23262 + 23266 + 23361 + 23362 + 23363 + 23372 *
  >>34 = 23211 + 23212 + 23213 *
  >>37 = 23241 + 23242 + 23247 *
  >>40 = 23231 + 23232 *
  >>43 = 23271 + 23273 + 23274 + 23275 *
  >>45 = 23281 + 23286 + 23287 + 23291 + 23293 + 23296 + 23297 + 23300 +
  >>    23311 + 23313 + 23314 + 23321 + 23323 + 23331 + 23334 + 23340 +
  >>    23353 + 23354 + 23355 *
  >>50 = 23351 + 23352 *
  >>71 = 23401 + 23402 + 23403 *
  >>55 = 23451 + 23452 + 23453 + 23454 *
  >>81 = 23501 + 23505 + 23510 + 23521 *
  >>64 = 23111 + 23608 *
  >>65 = 23112 + 23611 *
  >>74 = 23601 + 23603 + 23605 + 23604 + 23602 + 23613 + 23620 + 23631 +
  >>    23632 + 23633 + 23641 + 23642 *
  >>63 = 23651 + 23652 + 23655 + 23661 + 23662 + 23663 + 23670 *
  >>83 = 22704 *
  >>85 = 22950 + 23405 + 23551 + 23553 + 23502 + 23527 +
  >>    23700 + 23711 + 23713 + 23720 + 23730 + 23741 + 23742 + 23744 + 23745 +
  >>    23747 + 23748 + 23800 + 23852 + 23851 + 23853 + 23859 + 23900 +
  >>    23910 + 23921 + 23922 + 23926 + 23927 + 23930 +
  >>    26800 + 26851 + 26853 + 26854 + 26910 + 26921 + 26926 *
  >>"92S" = 24752 *
  >>"93S" = 24800 *
  >>"94S" = 24852 + 24851 + 24853 *
  >>"95S" = 24631 + 24632 + 24670 + 24730 + 24742 + 24745 + 24751 + 24921 *
  >>"93K" = 25800 *
  >>"94K" = 25851 + 25853 + 25854 *
  >>"95K" = 25410 + 25453 + 25751 + 25900 + 25921 *
  >>**

end;
```

```
/* lagmat.src - Macro som konverterer brukerens aggregeringsspesifikasjon, */  
/* og som produserer aggregeringsmatrisene. Dette er matriser */  
/* som bestaar av 1'ere og nuller, og selve aggregeringen utfoeres */  
/* ved aa multiplisere disse med de aktuelle datamatrisene. */  
/* Versjon for aa aggregere fra FNR til Modag i Regard */  
/* Programmert av Rune Johansen. Siste versjon 29 mai 1996 */
```

```
addfun main;  
procedure main()  
begin;  
  
    i=1;il=1;j=1;  
  
    GET type;  
  
    listefra = "lfra"||(type);  
    listetil = "ltil"||(type);  
  
    /* labelfilene leses inn */  
    lfra = getdata(listefra);  
    ltil = getdata(listetil);  
  
    n1 = nvals(lfra);  
    n2 = nvals(ltil);  
  
    matrise = crmat(n2,n1,0);  
  
    GET y;  
    while (y <> "***")  
    begin;  
        /* finner i hvilken rad neste 1-tall skal plasseres*/  
        i = datint(ltil,y);  
        GET y; /* likhetstegn,pluss eller stjerne */  
        while (y <> "**")  
        begin;  
            GET y;  
            /* finner hvilken kolonne 1-tallet skal plasseres */  
            j = datint(lfra,y);  
            /* plasserer et 1-tall i rad nr i og kolonne nr. j */  
            matrise=setrep(matrise,1,i,j);  
            va = values(matrise,i,j);  
            GET y;  
        end;  
        GET y;  
    end;  
  
    /* den konstruerte matrisen lagres paa fil */  
    utfil = "agg"||type;  
    PUTDATA(matrise,utfil,-1);  
  
end;
```

```
/* invest2.src - Bruttoinvesteringer i fast kapital etter sektor (JS) */
/* Eksogene størrelser. Lagres midlertidig. */
/* Programmert av Rune Johansen. Sist oppdatert 1.08.96 */

/* variable */
/* JS - bruttoinvesteringer etter sektor fordelt paa fylke */
/* FJS - fordelingsnøkkel beregnet på bakgrunn av FNR-data */
/* JKS - investeringer etter sektor (hentes fra Modag) */

addfun main;
procedure main()
begin;

  /* lister sektorer */
  liste = getdata("modag");
  nsekt = nvals(liste);
  /* fylkekoder */
  fylke = getdata("fylkkode");
  nf = nvals(fylke);

  /* sektorfordelte bruttoinvesteringer */
  /* fylkesvise investeringer i bedriftssektoren (unntatt sektorer */
  /* som i sin helhet er lagt til ekstrarfylket) beregnes som produktet */
  /* av eksogent gitte investeringer i hver sektor (JKS) og fylkesvise */
  /* fordelingsnøkler i basisåret (FJS) */

  fjs = getdata("fjs");
  for(i=1;i<=nf;i=i+1)
  begin; /* region */
    fy = values(fylke,i);
    for(j=1;j<=nsekt;j=j+1)
    begin; /* sektor */
      se = values(liste,j); /* sektorkode */
      fj = values(fjs,j,i);
      >>dofile utl_js&(se)&(fy) = &(fj)*jks&(se);
    end;
  end;

end;
```

```
/* lagqseq.src - Bruttoprodukt (QS) etter sektor og fylke. */
/* Implementert av Rune Johansen. Sist oppdatert 1.08.96 */

/* variable */
/* QS - bruttoprodukt etter sektor og fylke */
/* FQS - fordelingsnøkkel beregnet fra FNR */
/* Q - bruttorprodukt etter sektor (hentes fra Modag) */

addfun main;
procedure main()
begin;

  /* lister sektorer og fylker */
  sekt = getdata("modag");
  nse = nvals(sekt);
  fylke = getdata("fylkkode");
  nf = nvals(fylke);

  fqs = getdata("fqs");
  for(i=1;i<=nf;i=i+1)
  begin; /* fylke */
    fy = values(fylke,i);
    for(j=1;j<=nse;j=j+1)
    begin; /* sektor */
      se = values(sekt,j); /* sektorkode */
      fq = values(fqs,j,i);
      >>dofile ut1_qs&(se)&(fy) = &(fq)*q&(se);
    end;
  end;

end;
```

```
/* syss.src - Syssselsetning etter sektor og fylke */
/* Programmert av Rune Johansen. Sist oppdatert 5.2.97 */

/* variable */
/* LS - Syssselsetning etter sektor fordelt paa fylke */
/* FLS - fordelingsnøkkel beregnet fra FNR */
/* N - Syssselsetning etter sektor (hentes fra Modag) */

addfun main;
procedure main()
begin;

  /* lister sektorer og fylker */
  liste = getdata("modag");
  nsekt = nvals(liste);
  fylke = getdata("fylkkode");
  nf = nvals(fylke);

  fls = getdata("fls");
  for(i=1;i<=nf;i=i+1)
  begin; /* fylke */
    fy = values(fylke,i);
    for(j=1;j<=nsekt;j=j+1)
    begin; /* sektor */
      se = values(liste,j); /* sektorkode */
      fl = values(fl_s,j,i);
      >>do nn&(se)=1000*n&(se);
      /* trekker i fra utlendinger i utenriks sjofart i sektor 65 */
      if (se == "65") then
        >>do nn&(se)=nn&(se)-ut1_ls60u;
      /* trekker i fra vernepliktige i sektor 92s */
      if (se == "92S") then
        >>do nn&(se)=nn&(se)-1000*(ut1_nvpl16*nb1619+nvpl20*nb2024);
      >>dofile ut1_ls&(se)&(fy) = &(fl)*nn&(se);

    end;
  end;

end;
```

```
/* arbstyrk.src - Total arbeidsstyrke hentet fra MODAG */
/* Programmert av Rune Johansen. Sist oppdatert 3.10.96 */

/* variable */
/* NT - Total arbeidsstyrke (hentes fra Modag) */
/* NVPL - antall vernepliktige */
/* NS60U - antall utlendinger i utenriks sjofart */

addfun main;
procedure main()
begin;
  /* basisaar */
  get number aar;
  aar1 = convert(aar)||"A";

  /* Total arbeidsstyrke */
  /* Antall utlendinger i utenriks sjofart */
  >>dofile ut1_ns60u = (values(ut1_ls60u,&(aar1))/values(n65,&(aar1)))*n65;
  /* Maa trekke i fra vernepliktige og utlendinger i utenriks sjofart */
  >>dofile ut1_antot = 1000*nt - ut1_ns60u -
    >>1000*(ut1_nvpl16*nb1619 + nvpl20*nb2024);

end;
```

```
/* res.src - regner ut vekstprosenten fra et gitt aar til det neste */
/* for serier produsert av formodellen, dvs bruttoprodukt, investering */
/* og sysselsetting. Dersom observasjonen for det første året er lik 0 */
/* kan ikke vekstprosenten defineres medmindre observasjonen det andre */
/* året også er lik null. I så fall er vekstprosenten lik null. Hvis */
/* ikke returnerer macroen verdien 'NA', fordi den forsøker å dividere */
/* med 0. */
/* Programmert av Rune Johansen. Sist oppdatert 5.12.1997 */
```

```
addfun main;
procedure main()
begin;

  >>access inn type disk id ../ mode r;
  >>search data inn;
  liste = getdata("modag");
  nls = nvals(liste);
  liste2 = getdata("mod2");
  fylke = getdata("fylkkode");
  nf = nvals(fylke);
  fynavn = getdata("fylknavn");
  >>delsearch all;
  >>delaccess all;

  /* leser inn navn paa databasen der seriene skal hentes fra */
  /* (arbeidstdatabasen) */
  get navn " Navn paa arbeidsdatabase:";
  navn = lower(navn);

  /* leser inn fra-aar */
  get number aar0 "Tast inn aaret du skal regne vekst fra:";
  aar1 = convert(aar0) || "A";
  aar2 = aar0+1;
  aar3 = convert(aar2) || "A";

  /* landstall settes sammen i matriser, med sektorinndelingen loddrett */
  >>access ut1 type fame id /ssb/frisch/fame/regard/grlag/formod.db mode r;
  >>search data ut1;
  >>access inn type disk id ../ mode rw;

  print("Landstall");

  print("Bruttoprodukt");

  >>do lis2 = combine(
  for(i=1;i<=nls;i=i+1)
  begin;
    se2 = values(liste2,i);
    if (i<nls) then
      >>&(se2),
    else
      >>&(se2)
  end;
  >>);

  for(i=1;i<=nls;i=i+1)
  begin;
    se = values(liste,i);
    serie = "q" || (se);
    >>do inn_q&(se) = values(&(serie),&(aar1));
  end;

  >>do q1 = combine(
  for(i=1;i<=nls;i=i+1)
  begin;
    se = values(liste,i);
    serie = "inn_q" || (se);
    q = getdata(serie);
    if (i<nls) then
```

```

    >>&(q),
  else
    >>&(q)
end;
>>);

for(i=1;i<=nls;i=i+1)
begin;
  se = values(liste,i);
  serie = "q" || (se);
  >>do inn_q&(se) = values(&(serie),&(aar3));
end;

>>do q2 = combine(
for(i=1;i<=nls;i=i+1)
begin;
  se = values(liste,i);
  serie = "inn_q" || (se);
  q = getdata(serie);
  if (i<nls) then
    >>&(q),
  else
    >>&(q)
end;
>>);

>>do dq = q2-q1;
>>do bp = if dq==0 then 0 else 100*(q2-q1)/q1;

>>do bpl = partcomb(1,2,lis2,bp);
>>do prt.(bpl);

print("Investering");

for(i=1;i<=nls;i=i+1)
begin;
  se = values(liste,i);
  serie = "jks" || (se);
  >>do inn_jks&(se) = values(&(serie),&(aar1));
end;

>>do j1 = combine(
for(i=1;i<=nls;i=i+1)
begin;
  se = values(liste,i);
  serie = "inn_jks" || (se);
  jks = getdata(serie);
  if (i<nls) then
    >>&(jks),
  else
    >>&(jks)
end;
>>);

for(i=1;i<=nls;i=i+1)
begin;
  se = values(liste,i);
  serie = "jks" || (se);
  >>do inn_jks&(se) = values(&(serie),&(aar3));
end;

>>do j2 = combine(
for(i=1;i<=nls;i=i+1)
begin;
  se = values(liste,i);
  serie = "inn_jks" || (se);
  jks = getdata(serie);
  if (i<nls) then
    >>&(jks),

```



```
    else
        >>&(jks)
end;
>>);

>>do dj = j2-j1;
>>do jk = if dj==0 then 0 else 100*(j2-j1)/j1;

>>do jkl = partcomb(1,2,lis2,jk);
>>do prt.(jkl);

print("Sysselsetting");

for(i=1;i<=nls;i=i+1)
begin;
    se = values(liste,i);
    serie = "n"||(se);
    >>do inn_n&(se) = values(&(serie),&(aar1));
end;

>>do l1 = combine(
for(i=1;i<=nls;i=i+1)
begin;
    se = values(liste,i);
    serie = "inn_n"||(se);
    n = getdata(serie);
    if (i<nls) then
        >>&(n),
    else
        >>&(n)
end;
>>);

for(i=1;i<=nls;i=i+1)
begin;
    se = values(liste,i);
    serie = "n"||(se);
    >>do inn_n&(se) = values(&(serie),&(aar3));
end;

>>do l2 = combine(
for(i=1;i<=nls;i=i+1)
begin;
    se = values(liste,i);
    serie = "inn_n"||(se);
    n = getdata(serie);
    if (i<nls) then
        >>&(n),
    else
        >>&(n)
end;
>>);

>>do dl = l2-l1;
>>do lk = if dl==0 then 0 else 100*(l2-l1)/l1;
>>do lsl = partcomb(1,2,lis2,lk);
>>do prt.(lsl);

>>delsearch all;
>>delaccess all;

print("Fylkesvise tall");
>>access ut2 type fame id /ssb/frisch/fame/regard/data/&(navn).db mode r;
>>search data ut2;
>>access inn type disk id ../ mode rw;

print("Bruttoprodukt");
for(k=1;k<=nf;k=k+1)
```

```

begin;
  fy = values(fylke,k);

  for(i=1;i<=nls;i=i+1)
  begin;
    se = values(liste,i);
    serie = "qs" || (se) || (fy);
    >>do inn_qs&(se)&(fy) = values(&(serie),&(aar1));
  end;

  >>do q1 = combine(
  for(i=1;i<=nls;i=i+1)
  begin;
    se = values(liste,i);
    serie = "inn_qs" || (se) || (fy);
    qs = getdata(serie);
    if (i<nls) then
      >>&(qs),
    else
      >>&(qs)
  end;
  >>);

  for(i=1;i<=nls;i=i+1)
  begin;
    se = values(liste,i);
    serie = "qs" || (se) || (fy);
    >>do inn_qs&(se)&(fy) = values(&(serie),&(aar3));
  end;

  >>do q2 = combine(
  for(i=1;i<=nls;i=i+1)
  begin;
    se = values(liste,i);
    serie = "inn_qs" || (se) || (fy);
    qs = getdata(serie);
    if (i<nls) then
      >>&(qs),
    else
      >>&(qs)
  end;
  >>);

  >>do dq = q2-q1;
  >>do q3 = if dq==0 then 0 else 100*(q2-q1)/q1;

  >>do bp&(fy) = partcomb(1,2,lis2,q3);
  fn = values(fynavn,k);
  print(fn);
  >>do prt.(bp&(fy));
end;

print("Investering");
for(k=1;k<=nf;k=k+1)
begin;
  fy = values(fylke,k);

  for(i=1;i<=nls;i=i+1)
  begin;
    se = values(liste,i);
    serie = "js" || (se) || (fy);
    >>do inn_js&(se)&(fy) = values(&(serie),&(aar1));
  end;

  >>do j1 = combine(
  for(i=1;i<=nls;i=i+1)
  begin;
    se = values(liste,i);
    serie = "inn_js" || (se) || (fy);

```

```

    js = getdata(serie);
    if (i<nls) then
        >>&(js),
    else
        >>&(js)
end;
>>);

for(i=1;i<=nls;i=i+1)
begin;
    se = values(liste,i);
    serie = "js" || (se) || (fy);
    >>do inn_js&(se)&(fy) = values(&(serie),&(aar3));
end;

>>do j2 = combine(
for(i=1;i<=nls;i=i+1)
begin;
    se = values(liste,i);
    serie = "inn_js" || (se) || (fy);
    js = getdata(serie);
    if (i<nls) then
        >>&(js),
    else
        >>&(js)
end;
>>);

>>do dj = j2-j1;
>>do j3 = if dj==0 then 0 else 100*(j2-j1)/j1;

>>do jk&(fy) = partcomb(1,2,lis2,j3);
fn = values(fynavn,k);
print(fn);
>>do prt.(jk&(fy));
end;

print("Sysselsetting");
for(k=1;k<=nf;k=k+1)
begin;
    fy = values(fylke,k);

    for(i=1;i<=nls;i=i+1)
begin;
    se = values(liste,i);
    serie = "ls" || (se) || (fy);
    >>do inn_ls&(se)&(fy) = values(&(serie),&(aar1));
end;

>>do l1 = combine(
for(i=1;i<=nls;i=i+1)
begin;
    se = values(liste,i);
    serie = "inn_ls" || (se) || (fy);
    ls = getdata(serie);
    if (i<nls) then
        >>&(ls),
    else
        >>&(ls)
end;
>>);

for(i=1;i<=nls;i=i+1)
begin;
    se = values(liste,i);
    serie = "ls" || (se) || (fy);
    >>do inn_ls&(se)&(fy) = values(&(serie),&(aar3));
end;

```

```
>>do l2 = combine(
for(i=1;i<=nls;i=i+1)
begin;
  se = values(liste,i);
  serie = "inn_ls"||(se)||(fy);
  ls = getdata(serie);
  if (i<nls) then
    >>&(ls),
  else
    >>&(ls)
end;
>>);

>>do dl = l2-l1;
>>do l3 = if dl==0 then 0 else 100*(l2-l1)/l1;

>>do ls&(fy) = partcomb(1,2,lis2,l3);
fn = values(fynavn,k);
print(fn);
>>do prt.(ls&(fy));
end;

end;
```

```

/* yprmod2.src - Beregner yrkesprosentene. Eksogen versjon. */
/* Oversatt til Potroll av Rune Johansen. Sist oppdatert 5.2.97 */

/* y2m hentes fra MOSART */
/* med utgangspunkt i yrkesprosentene for aaret foer forste simuleringsaar */
/* skal yrkesprosentene endres med samme differanse for hvert aar som */
/* endringene i MOSART-tallene. */
/* Yrkesprosentene trengs for beregning av arbeidsstyrke i modellen ARBEID */

addfun main;
procedure main()
begin;

  /* henter inn forste og siste simuleringsaar */
  get number aar1;
  get number aar2;

  for(i=1;i<=2;i=i+1)
  begin; /* kjonn */
    for(l=3;l<=9;l=l+1)
    begin; /* aldersgruppe 3-9 (16-74 aar) */
      for(j=1;j<=3;j=j+1)
      begin; /* utdanning */

        aar3 = aar1-1;
        a1 = convert(aar1)||"A";
        a3 = convert(aar3)||"A";
        /* endringen i MOSART-tallene */
        >>do diff = values(ut1_y2m&(i)&(l)&(j),&(a1)) -
        >>          values(ut1_y2m&(i)&(l)&(j),&(a3));
        /* yrkesprosenten oppdateres */
        >>dofile ut1_ypr&(i)&(l)&(j)=
        >>reshape(values(ut1_yp&(i)&(l)&(j),&(a3))+diff,&(a1));

        for(ar=aar1+1;ar<=aar2;ar=ar+1)
        begin;
          ar3 = ar-1;
          a1 = convert(ar)||"A";
          a3 = convert(ar3)||"A";
          /* endringen i MOSART-tallene */
          >>do diff = values(ut1_y2m&(i)&(l)&(j),&(a1)) -
          >>          values(ut1_y2m&(i)&(l)&(j),&(a3));
          /* yrkesprosentene oppdateres */
          >>dofile ut1_ypr&(i)&(l)&(j)=
          >>overlay(ut1_ypr&(i)&(l)&(j),
          >>reshape(values(ut1_ypr&(i)&(l)&(j),&(a3))+diff,&(a1)));

        end;
      end;
    end;
  end;

end;

```

Macroer som genererer likninger i
ARBEID

```
/* arbeid.src - macro for aa lage hoveddelen av regard. Genererer */
/* likningene i modellen ARBEID. */
/* Programmert av Rune Johansen. Sist oppdatert 14.11.1996 */

addfun main;
procedure main()
begin;

    /* aksesserer og søker paa databasen med koeffisienter */
    >>access koef type trolldb id ./koeff.tr1 mode r;
    >>search data koef;

    >>usemod arbeid;modedit;

    /* likninger for sammensetningen av sysselsetning etter kjonn */
    /* og utdanning */
    &nysyssutd;

    /* etterspørsel etter arbeid. Lager ogsaa likninger for syss for */
    /* ekstrasfylket */
    &ettersp2;

    /* sysselsetningen for hver region er den samme enten man splitter */
    /* opp etter sektor eller kjonn/utdanning. */
    &gam2;

    /* fremskrivning av personer etter utdanning paa gr1.nivaa */
    &nyutd;

    /* flyttelikninger */
    /* flytting etter personkjennetegn */
    &flyttmod;
    /* generelle flyttelikninger */
    &flytmod2;
    /* flytting av barn
    &flytmod5;

    /* inn- og utvandring */
    &innutv;

    /* nye likninger for sysselsatte */
    &nysyss;

    /* Likningene for yrkesprosent, tidligere laget ved et kall paa */
    /* macroen yprmod, utgaar fordi yrkesprosenten naa blir beregnet */
    /* eksogent ved hjelp av MOSART-tall (i formodellen). */

    /* tilbud av arbeid (uten flytting) */
    &nytilbud;

    /* De genererte likningene lagres paa modellen ARBEID */
    >>filemod;

end;
```

```

/* nysyssutd.src - lager nye likninger for sammensetningen av sysselsetn. */
/* etter kjonn og utdanning. */
/* Oversatt til Potroll av Rune Johansen. Sist oppdatert 3.10.96 */

/* koeffisienter */
/* NYUT - sysselsetningsandeler etter sektor for hver region, kjonn, utd */

/* variable */
/* L - endogen sysselsetting etter kjonn, utdanning og region */
/* LS - Sysselsetting etter sektor og region */
/* MGAM - endogen skaleringsfaktor etter region, sikrer at den totale */
/*      sysselsettingen er lik uansett hvordan den splittes opp */
/* GAMMA - eksogen korreksjonsfaktor (trendledd) som justerer syssel- */
/*      etter kjonn, utdanning, region over tid. */

addfun main;
procedure main()
begin;

  /* sektorlisten leses inn */
  sektor = getdata("modag");
  nr = nvals(sektor);

  >>addeq bottom
  /* lokke over kjonn */
  for(i=1;i<=2;i=i+1)
  begin;
    /* lokke over utdanningskategorier */
    for(j=1;j<=3;j=j+1)
    begin;
      /* lokke over region */
      for(k=1;k<=7;k=k+1)
      begin;
        ii = convert(i); jj = convert(j); kk = convert(k);
        koeff = "nyut" || (ii) || (jj) || (kk);
        nyut = getdata(koeff);
        flagg = 0;
        >>l&(i)&(j)&(k)'n=mgam&(k)'n*gamma&(i)&(j)&(k)*(
        for(m=1;m<=nr;m=m+1)
        begin;
          re = values(sektor,m);
          if (re <> "89") then
          begin;
            if (flagg == 0) then
              ny = values(nyut,m);
            else
              ny = values(nyut,m-1);
            if (ny <> 0) then
              >>&(ny)*ls&(re)&(k)+
          end; else
            flagg = 1;
          end;
        >>0),
      end;
    end;
  end;
end;
>>;
end;

```



```

/* ettersp2.src - ettersporsel etter arbeid */
/* Oversatt til Potroll av Rune Johansen. Sist oppdatert 4.7.96 */

/* koeffisienter */
/* LPERS - sysselsettingsandeler for ekstrasfylket. tilsv. NYUT */
/* NPEND - netto pendling etter kjonn og utdanning */

/* variable */
/* L - endogen sysselsetting etter kjonn, utdanning og region */
/* LS - sysselsetting etter sektor og region */
/* NP - endogen netto pendling etter kjonn, utdanning og region */
/* JNP - endogen justeringsfaktor for pendling etter kjonn og utdanning */
/* XML - endogen markedsleieindikator etter kjonn, utdanning og region */
/* AS - endogen arbeidsstyrke etter kjonn, alder, utdanning og region */

addfun main;
procedure main()
begin;

  /* sortert liste over prod.sekt. i regard */
  sekt = getdata("modag");
  nsek = nvals(sekt);

  >>addeq bottom
  for(i=1;i<=2;i=i+1)
  begin; /* lokke paa kjonn */
    for(j=1;j<=3;j=j+1)
    begin; /* utdanning */
      k=8; /* likninger for ekstrasfylket */
      ii = convert(i); jj = convert(j);
      /* persongruffordeling */
      koeff = "lpers" || (ii) || (jj) || "8";
      lpers = getdata(koeff);
      >>l&(i)&(j)&(k) 'n=mgam&(k) 'n*gamma&(i)&(j)&(k) * (
        for(m=1;m<=nsek;m=m+1)
        begin;
          se = values(sekt,m);
          lp = values(lpers,m);
          if (lp <> 0) then
            >>&(lp)*ls&(se)&(k)+
          end;
        >>0),
      end;
    end;
  end;

  /* nettopendling for innenlandske regioner */

  for(i=1;i<=2;i=i+1)
  begin; /* lokke paa kjonn */
    for(j=1;j<=3;j=j+1)
    begin; /* utdanning */
      ii = convert(i); jj = convert(j);
      koeff = "npend" || (ii) || (jj);
      npend = getdata(koeff);
      for(k=1;k<=7;k=k+1)
      begin; /* landsdel (innenlands) */
        np = values(npend,k);
        >>np&(i)&(j)&(k) 'n=
        if (np > 1) then
          >>(1-&(np)) * jnp&(i)&(j) 'n*1&(i)&(j)&(k) 'n
        else
          >>(1-&(np)) * 1&(i)&(j)&(k) 'n
          >>+s.np&(i)&(j)&(k),
        end;
      /* konsistens netto innenlandsk pendling */
      /* lokke paaa region */
      for(k=1;k<=7;k=k+1)
      begin;
        if (k == 1) then

```

```
        >>np&(i)&(j)&(k)
    else
        >>+np&(i)&(j)&(k)
end;
>>=0,
/* beregning av markedsleie */
ii = convert(i); jj = convert(j);
koeff="ekstra"||(ii)||(jj);
ekstra = getdata(koeff);
/* lokke region */
for(k=1;k<=7;k=k+1)
begin;
    x = values(ekstra,k);
    >>xml&(i)&(j)&(k)'n=(l&(i)&(j)&(k)'n-np&(i)&(j)&(k)'n
    if (x <> 0) then
        >>+&(x)*l&(i)&(j)8'n
    >>)/(
    /* lokke aldersgrupper, kun gruppe 3-9 i arbeid*/
    for(l=3;l<=9;l=l+1)
    begin;
        >>as&(i)&(l)&(j)&(k)'n+
    end;
    >>0),
end;
end;
end;

>>;
end;
```

```
/* gam2.src - sysselsetningen i hver region er den samme enten */
/* man splitter opp etter sektor eller kjonn/utdanning. */
/* Oversatt til Potroll av Rune Johansen. Sist oppdatert 4.7.96 */

/* variable */
/* L - Sysselsetting etter kjonn, utdanning og region */
/* LS - Sysselsetting etter sektor og region */

addfun main;
procedure main()
begin;

    sekt1 = getdata("modag");
    nv = nvals(sekt1);

    >>addeg bottom
    /* lokke over region (inkl ekstrarfylket) */
    for(k=1;k<=8;k=k+1)
    begin;
        /* lokke over sektorliste */
        for(m=1;m<=nv;m=m+1)
        begin;
            va = values(sekt1,m);
            if (va <> "89") then
            begin;
                if (m == 1) then
                    >>ls&(va)&(k)
                else
                    >>+ls&(va)&(k)
            end;
        end;
    end;
    >>=
    /* kjonn */
    for(i=1;i<=2;i=i+1)
    begin;
        /* utdanningskategori */
        for(j=1;j<=3;j=j+1)
        begin;
            >>l&(i)&(j)&(k)+
        end;
    end;
    >>0,
end;
>>;
end;
```

```

/* nyutd.src - fremskrivning av personer etter utdanning paa gr.nivaa */
/* Oversatt til Potroll av Rune Johansen. Sist oppdatert 4.7.96 */

/* koeffisienter */
/* UTIND - indekser for vekst i utdanning */
/* UTFORD - fordeling av personer med hoyere utdanning (3) */
/*      i ald.gr 3 (16-19) */

/* variable */
/* B - Folketall etter kjonn, alder, utdanning og region */
/* BUTG (1) - endogent folketall etter kjonn, alder, utdanning og region */
/*      ved utgangen av aaret */
/* BUTG (2) - Folketall etter kjonn, alder og region ved utg. av aaret */
/*      (aggregert over utdanningskategorier). Eksogen variabel */
/*      beregnet i demografidelen. */
/* M - vekstrater for personer i aldersgrupper fra Mosart */
/* JB - endogen justering av utdanningsfordeling */
/* BM - absolutt tilveskt i aldersgruppe 3 hele landet */

/* Den endogent beregnede variabelen BUTG, beregnet av likningene generert */
/* av denne macroen, blir ved utgangen av aaret justert i samsvar med den */
/* eksogene BUTG beregnet i demografidelen (UTBEF). Denne forelopige */
/* eksogene variabelen omfatter ikke inn- og utvandring samt flytting. */

addfun main;
procedure main()
begin;

    >>addeq bottom

    /* befolkning etter kjonn, alder, utdanning og region */
    /* lokke over kjonn */
    for(i=1;i<=2;i=i+1)
    begin;
        /* lokke over utdanningskategori */
        for(j=1;j<=3;j=j+1)
        begin;
            /* aldersgrupper 4-9 (20-74 aar) */
            for(l=4;l<=9;l=l+1)
            begin;
                ii = convert(i); jj = convert(j); ll = convert(l);
                koeff = "utind"||(ii)||"3"||(jj);
                utind = getdata(koeff); /* indekser vekst utdanning */
                for(k=1;k<=7;k=k+1)
                begin; /* region */
                    ut = values(utind,k);
                    >>butg&(i)&(l)&(j)&(k)'n=
                    if (ut <> 0) then
                    begin;
                        >>&(ut)*m&(i)&(l)&(j)*jb&(i)&(l)&(k)'n*
                        >>b&(i)&(l)&(j)&(k)-
                    end;
                    /* tallene justeres ved aa trekke i fra netto */
                    /* innvandring, fordi MOSART-tallene for befolknings- */
                    /* vekst inkluderer denne, mens REGARD behandler denne */
                    /* separat */
                    >>inv&(i)&(l)&(k)'n+utv&(i)&(l)&(k)'n,
                end;
            end;
        end;
    end;

    /* de fleste i aldersgruppe 3 (16-19) har IKKE hoyere utdanning */
    for(j=1;j<=2;j=j+1)
    begin;
        ii = convert(i); jj = convert(j);
        koeff = "utind"||(ii)||"3"||(jj);
        utind = getdata(koeff);
        /* lokke over region */
        for(k=1;k<=7;k=k+1)
        begin;

```

```

        ut = values(utind,k);
        >>butg&(i)3&(j)&(k)'n=
        if (ut <> 0) then
            >>&(ut)*m&(i)3&(j)*jb&(i)3&(k)'n*b&(i)3&(j)&(k)-
        >>inv&(i)3&(k)'n+utv&(i)3&(k)'n,
    end;
end;
/* spesiallosning for de faa i ald.gr.3 (16-19) som har hoyere utd. */
/* fordelingen av gruppe 3,3 */
ii = convert(i);
koeff = "utford"||(ii);
utford = getdata(koeff);
/* lokke over region */
for(k=1;k<=7;k=k+1)
begin;
    ut = values(utford,k);
    >>butg&(i)33&(k)'n=
    if (ut <> 0) then
        >>&(ut)*bm&(i)33,
    else
        >>0,
end;
/* lokke over aldersgruppe 3-9 (16-74 aar) */
for(l=3;l<=9;l=l+1)
begin;
    /* lokke over region */
    for(k=1;k<=7;k=k+1)
    begin;
        /* utdanningskategori */
        for(j=1;j<=3;j=j+1)
        begin;
            if (j == 1) then
                >>butg&(i)&(l)&(j)&(k)
            else
                >>+butg&(i)&(l)&(j)&(k)
            end;
            >>=butg&(i)&(l)&(k),
        end;
    end;
end;
end;
>>
end;

```

```

/* flyttmod.src - Flytting etter personkjennetegn */
/* Oversatt til Potroll av Rune Johansen. Sist oppdatert 6.9.96 */

/* koeffisienter */
/* FLYT - flytterater for aldersgruppe 3 (16-19) hoy utdanning (3) */
/* FLP - estimerte parametre i flyttemodellen */

/* variable */
/* UT - utflyttere etter kjonn, alder, utdanning, fraflyttingsregion */
/* UTB - utflyttede barn */
/* IN - innflyttere til regionen etter kjonn, alder, utdanning */
/* INB - innflyttede barn */
/* B - folketall */
/* S.UTB - eksogen korreksjon barn, utflytting */
/* S.INB - eksogen korreksjon barn, innflytting */
/* S.UT - eksogen tilpasning basisaar utflytting inkl barn */
/* S.IN - eksogen tilpasning basisaar innflytting inkl barn */
/* XML - markedsleie */

addfun main;
procedure main()
begin;

  >>addeq bottom

  /* beregning av flytting for personer med utdanningsfordeling */

  /* aldersgruppe 3 (16-19) og utdanning 3 (hoyere), spesialbehandling */
  for(i=1;i<=2;i=i+1)
  begin; /* kjonn */
    for(k=1;k<=7;k=k+1)
    begin; /* fra- og tilflyttingsregion */
      ii = convert(i); kk = convert(k);
      koeff = "flyt" || (ii) || "33" || (kk);
      flytt = getdata(koeff);
      >>ut&(i)33&(k)'n=(
        for(k1=1;k1<=7;k1=k1+1)
        begin; /* tilflyttingsregion */
          fl = values(flytt,k1);
          >>&(fl)+
        end;
      >>0)*b&(i)33&(k)/1000+s.ut&(i)33&(k),
      >>in&(i)33&(k)'n=
      for(k1=1;k1<=7;k1=k1+1)
      begin; /* fraflyttingsregion */
        fl = values(flytt,k1);
        if (fl <> 0) then
          >>&(fl)*b&(i)33&(k1)/1000+
        end;
      >>s.in&(i)33&(k),
    end;
  end;

  /* generelt tilfelle, aldersgruppe 7-9 (45-74) */

  for(i=1;i<=2;i=i+1)
  begin; /* kjonn */
    for(k=1;k<=7;k=k+1)
    begin; /* fraflyttingsregion */
      for(l=7;l<=9;l=l+1)
      begin; /* aldersgruppe */
        for(j=1;j<=3;j=j+1)
        begin; /* utdanning */
          ii = convert(i); jj = convert(j);
          kk = convert(k); ll = convert(l);
          koeff = "flyt" || (ii) || (ll) || (jj) || (kk);
          flytt = getdata(koeff);
          >>ut&(i)&(l)&(j)&(k)'n=(

```

```

        for(k1=1;k1<=7;k1=k1+1)
        begin; /* fraflyttingregion */
            fl = values(flytt,k1);
            >>&(fl)+
        end;
        >>0)*b&(i)&(l)&(j)&(k)/1000+s.ut&(i)&(l)&(j)&(k),
        >>in&(i)&(l)&(j)&(k)'n=
        for(k1=1;k1<=7;k1=k1+1)
        begin; /* fraflyttingsregion */
            fl = values(flytt,k1);
            if (fl <> 0) then
                >>&(fl)*b&(i)&(l)&(j)&(k1)/1000+
            end;
            >>s.in&(i)&(l)&(j)&(k),
        end;
    end;
end;

/* utflytting for andre enn ungdommer med hoyeste utdanning */
/* generelt tilfelle aldersgruppe 4-6 (20-44) */

for(i=1;i<=2;i=i+1)
begin; /* kjonn */
    for(k=1;k<=7;k=k+1)
    begin; /* fraflyttingsregion */
        for(l=4;l<=6;l=l+1)
        begin; /* aldersgrupper */
            for(j=1;j<=3;j=j+1)
            begin; /* utdanning */
                ii = convert(i); jj = convert(j);
                kk = convert(k); ll = convert(l);
                koeff = "flp"||ii||ll||jj||kk);
                flytt = getdata(koeff);
                >>ut&(i)&(l)&(j)&(k)'n=b&(i)&(l)&(j)&(k)*((
                for(k1=1;k1<=6;k1=k1+1)
                begin; /* tilflyttingsregion */
                    fl = values(flytt,k1);
                    if (k1<k) then
                        if (fl <> 0) then
                            begin;
                                >>exp(&(fl)*xml&(i)&(j)&(k)/xml&(i)&(j)&(k1))+
                            end; else
                                >>1+
                            else
                                begin;
                                    k2=k1+1;
                                    if (fl<>0) then
                                        begin;
                                            >>exp(&(fl)*xml&(i)&(j)&(k)/xml&(i)&(j)&(k2))+
                                        end; else
                                            >>1+
                                end;
                            end;
                >>0)/(
                for(k1=1;k1<=6;k1=k1+1)
                begin; /* tilflyttingsregion */
                    fl = values(flytt,k1);
                    if (k1<k) then
                        if (fl<>0) then
                            begin;
                                >>exp(&(fl)*xml&(i)&(j)&(k)/xml&(i)&(j)&(k1))+
                            end; else
                                >>1+
                            else
                                begin;
                                    k2=k1+1;
                                    if (fl<>0) then
                                        begin;

```

```

        >>exp(&(fl)*xml&(i)&(j)&(k)/xml&(i)&(j)&(k2))+
        end; else
        >>1+
    end;
end;
    >>1))+s.ut&(i)&(l)&(j)&(k),
end;
end;

/* normalt tilfelle aldersgruppe 3, ingen har hoyere utdanning */
for(j=1;j<=2;j=j+1)
begin; /* utdanning */
    ii = convert(i); jj = convert(j); kk = convert(k);
    koeff = "flp"||"(ii)||"3"||"(jj)||"(kk);
    flytt = getdata(koeff);
    >>ut&(i)3&(j)&(k)'n=b&(i)3&(j)&(k)*((
    for(k1=1;k1<=6;k1=k1+1)
    begin; /*tilflyttingsregion */
        fl = values(flytt,k1);
        if (k1<k) then
            if (fl <> 0) then
                begin;
                    >>exp(&(fl)*xml&(i)&(j)&(k)/xml&(i)&(j)&(k1))+
                    end; else
                    >>1+
                else
                begin;
                    k2=k1+1;
                    if (fl<>0) then
                        begin;
                            >>exp(&(fl)*xml&(i)&(j)&(k)/xml&(i)&(j)&(k2))+
                            end; else
                            >>1+
                        end;
                    end;
                >>0)/(
                for(k1=1;k1<=6;k1=k1+1)
                begin; /* tilflyttingsregion */
                    fl = values(flytt,k1);
                    if (k1<k) then
                        if (fl<>0) then
                            begin;
                                >>exp(&(fl)*xml&(i)&(j)&(k)/xml&(i)&(j)&(k1))+
                                end; else
                                >>1+
                            else
                            begin;
                                k2=k1+1;
                                if (fl<>0) then
                                    begin;
                                        >>exp(&(fl)*xml&(i)&(j)&(k)/xml&(i)&(j)&(k2))+
                                        end; else
                                        >>1+
                                    end;
                                end;
                            end;
                            >>1))+s.ut&(i)3&(j)&(k),
                        end;
                    end;
                end;
            end;
        end;
    >>;
end;

```



```

/* flytmod2.src - Likninger for innflytting til hver region. */
/* Oversatt til Potroll av Rune Johansen. Sist oppdatert 5.7.96 */

/* koeffisienter */
/* FLP - flyttematriser */

/* variable */
/* IN - innflytting etter kjonn, alder, utdanning for hver region */
/* B - folketall etter kjonn, alder, utdanning og fraflyttingsregion */
/* XML - markedsleie etter kjonn, utdanning og region */

/* NB!! Maa beregne flytteandeler paa nytt, da utflytting fra en gitt */
/* region til en annen i macroen flyttmod ikke blir beregnet spesifikt. */

addfun main;
procedure main()
begin;

  >>addeq bottom

  /* innflytting */

  for(i=1;i<=2;i=i+1)
  begin; /* kjonn */
    for(l=4;l<=6;l=l+1)
    begin; /* aldersgruppene 4-6 (20-44 aar) */
      for(j=1;j<=3;j=j+1)
      begin; /* utdanning */
        for(k=1;k<=7;k=k+1)
        begin; /*innflyttingsregion */
          >>in&(i)&(l)&(j)&(k)'n=
          for(k1=1;k1<=6;k1=k1+1)
          begin; /* fraflyttingsregion. Alle andre enn 1 */
            if (k1<k) then
            begin;
              k2=k1;k3=1;
            end; else
            begin;
              k2=k1+1;k3=0;
            end;
            ii = convert(i); ll = convert(l);
            jj = convert(j); mm = convert(k2);
            koeff = "flp" || (ii) || (ll) || (jj) || (mm);
            flytt = getdata(koeff);
            k4 = k - k3;
            fl = values(flytt,k4);
            >>b&(i)&(l)&(j)&(k2)
            /* beregner utflyttingsandel til region k */
            if (fl <> 0) then
              >>*exp(&(fl)*xml&(i)&(j)&(k2)/xml&(i)&(j)&(k))
            >>/ (
            for(k5=1;k5<=6;k5=k5+1)
            begin;
              if (k5<k2) then
                k6=k5;
              else
                k6=k5+1;
              fl = values(flytt,k5);
              if (fl <> 0) then
                >>exp(&(fl)*xml&(i)&(j)&(k2)/xml&(i)&(j)&(k6))+
            end;
            >>1)+
          end;
        end;
      end;
    end;
  end;
  >>s.in&(i)&(l)&(j)&(k),
end;
end;
end;
/* behandling av aldersgruppe 3 UTEN hoyere utdanning */
for(j=1;j<=2;j=j+1)

```

```

begin; /* utdanning */
  for(k=1;k<=7;k=k+1)
    begin; /* innflyttingsregion */
      >>in&(i)3&(j)&(k)'n=
        for(k1=1;k1<=6;k1=k1+1)
          begin;
            if (k1<k) then
              begin;
                k2=k1;k3=1;
              end; else
              begin;
                k2=k1+1;k3=0;
              end;
            ii = convert(i); jj = convert(j);
            mm = convert(k2);
            koeff = "flp" || (ii) || "3" || (jj) || (mm);
            flytt = getdata(koeff);
            k4 = k - k3;
            fl = values(flytt,k4);
            >>b&(i)3&(j)&(k2)
            if (fl <> 0) then
              >>*exp(&(fl)*xml&(i)&(j)&(k2)/xml&(i)&(j)&(k))
            >>/ (
            for(k5=1;k5<=6;k5=k5+1)
              begin;
                if (k5<k2) then
                  k6=k5;
                else
                  k6=k5+1;
                fl = values(flytt,k5);
                if (fl <> 0) then
                  >>exp(&(fl)*xml&(i)&(j)&(k2)/xml&(i)&(j)&(k6))+
                end;
              >>1)+
            end;
            >>s.in&(i)3&(j)&(k),
          end;
        end;
      end;
    end;
  >>;
end;

```

```

/* flytmod5.src - Likninger for flytting av barn */
/* Oversatt til Potroll av Rune Johansen. Sist oppdatert 5.7.96 */

/* koeffisienter */
/* BFLIN - flytterater barn */
/* FLP - flytterater voksne */

/* variable */
/* FLRB - Flytterate for barn fra region til region */

addfun main;
procedure main()
begin;

  >>addeq bottom

  for(i=1;i<=2;i=i+1)
  begin; /* kjonn */
    for(l=1;l<=2;l=l+1)
    begin; /* aldersgruppe (barn) */
      lv = l + 4; /* aldersgruppe 'moedregenerasjon' */
      for(k=1;k<=7;k=k+1)
      begin; /* fraflyttingsregion */
        ii = convert(i); ll = convert(l); kk = convert(k);
        koeff = "bflin" || (ii) || (ll) || (kk);
        bflin = getdata(koeff);
        for(k1=1;k1<=7;k1=k1+1)
        begin; /* tilflyttingsregion */
          if (k1 <> k) then
          begin;
            >>flrb&(i)&(l)&(k)&(k1)'n=
            bf = values(bflin,k1);
            if (k1<k) then
              k2=k1;
            else
              k2=k1-1;
            if (bf <> 0) then
            begin;
              >>&(bf)* (
              for(j=1;j<=3;j=j+1)
              begin; /* utdanning voksne kvinner ('moedre') */
                ll = convert(lv); jj = convert(j);
                kk = convert(k);
                koeff = "flp2" || (ll) || (jj) || (kk);
                flp = getdata(koeff);
                fl = values(fl, k2);
                >>b2&(lv)&(j)&(k)
                if (k1<k) then
                begin;
                  if (fl <> 0) then
                    >>*exp(&(fl)*xml2&(j)&(k)/xml2&(j)&(k1))
                  else
                    >>1
                end; else
                begin;
                  if (fl <> 0) then
                    >>*exp(&(fl)*xml2&(j)&(k)/xml2&(j)&(k2))
                  else
                    >>1
                end;
              >>/ (
              for(k3=1;k3<=6;k3=k3+1)
              begin;
                fl = values(fl, k3);
                if (k3<k) then
                begin;
                  if (fl <> 0) then
                    >>exp(&(fl)*xml2&(j)&(k)/xml2&(j)&(k3))+

```

```

        else
            >>1+
        end; else
        begin;
            k4 = k3+1;
            if (f1 <> 0) then
                >>exp(&(f1)*xml2&(j)&(k)/xml2&(j)&(k4))+
            else
                >>1+
            end;
        end;
        >>1)+
    end;
    >>0)/(
    for(j=1;j<=3;j=j+1)
    begin;
        >>b2&(lv)&(j)&(k)+
    end;
    >>0),
end; else
    >>0,
    end; /* slutt test k1<>k */
end; /* slutt k1-lokke */
end; /* slutt k-lokke */
/* relasjoner for utflytting for barn */
for(k=1;k<=7;k=k+1)
begin; /* fraflyttingsregion */
    >>utb&(i)&(l)&(k)'n=barn&(i)&(l)&(k)*(
    for(k1=1;k1<=6;k1=k1+1)
    begin;
        if (k1<k) then
            begin;
                k2=k1;k3=1;
            end; else
            begin;
                k2=k1+1;k3=0;
            end;
            >>flrb&(i)&(l)&(k)&(k2)+
        end;
        >>0)+s.utb&(i)&(l)&(k),
    end;
end;
end;

/* innflytting barn */

for(i=1;i<=2;i=i+1)
begin; /* kjonn */
    for(l=1;l<=2;l=l+1)
    begin; /* aldersgruppe 1-2 (barn 0-15 aar) */
        for(k=1;k<=7;k=k+1)
        begin; /* innflyttingsregion */
            >>inb&(i)&(l)&(k)'n=
            for(k1=1;k1<=6;k1=k1+1)
            begin;
                if (k1<k) then
                    k2 = k1;
                else
                    k2 = k1 + 1;
                >>barn&(i)&(l)&(k2)*flrb&(i)&(l)&(k2)&(k)+
            end;
            >>s.inb&(i)&(l)&(k),
        end;
    end;
end;
end;

>>;
end;
```

```

/* innutv.src - Beregning av inn- og utvandring */
/* Oversatt til Potroll av Rune Johansen. Sist oppdatert 9.7.96 */

/* koeffisienter */
/* UTVAN - fordeling av utvandring paa kjonn og alder */
/* INVAN - fordeling av innvandring paa kjonn og alder */

/* Variable */
/* INV - innvandrng etter kjonn alder og region */
/* UTV - utvandring etter kjonn alder og region */
/* B - befolkning etter kjonn alder utdanning og region */
/* BARN - barnetall */

addfun main;
procedure main()
begin;

  >>addeq bottom

  for(i=1;i<=2;i=i+1)
  begin; /* kjonn */
    for(l=3;l<=9;l=l+1)
    begin; /* aldersgruppe 3-9 (16-74 aar) */
      ii = convert(i); ll = convert(l);
      koeff = "utvan" || (ii) || (ll);
      utvand = getdata(koeff);
      for(k=1;k<=7;k=k+1)
      begin; /* region */
        ut = values(utvand,k);
        >>utv&(i)&(l)&(k) 'n=
        /* aggregerer befolkning over utdanning */
        if (ut <> 0) then
          >>&(ut) * (b&(i)&(l)1&(k)+b&(i)&(l)2&(k)+b&(i)&(l)3&(k)) / 1000,
        else
          >>0,
        end;
      end;
    end;

  for(l=1;l<=2;l=l+1)
  begin; /* aldersgruppe 1 og 2 (0-15 aar), barn */
    ii = convert(i); ll = convert(l);
    koeff = "utvan" || (ii) || (ll);
    utvand = getdata(koeff);
    for(k=1;k<=7;k=k+1)
    begin;
      ut = values(utvand,k);
      >>utv&(i)&(l)&(k) 'n=
      if (ut <> 0) then
        >>&(ut) * barn&(i)&(l)&(k) / 1000,
      else
        >>0,
      end; /* slutt k-lokke */
    end;
  end;

  /* beregning av innvandring */

  >>bruttoin'n=
  for(i=1;i<=2;i=i+1)
  begin; /* kjonn */
    for(k=1;k<=7;k=k+1)
    begin; /* region */
      for(l=1;l<=9;l=l+1)
      begin; /* aldersgrupper (0-74 aar) */
        >>utv&(i)&(l)&(k)+
      end;
    end;
  end;
  >>nettoin,

```

```
for(i=1;i<=2;i=i+1)
begin; /* kjonn */
  for(l=1;l<=9;l=l+1)
  begin; /* alder */
    ii = convert(i); ll = convert(l);
    koeff = "invan" || (ii) || (ll);
    innv = getdata(koeff);
    for(k=1;k<=7;k=k+1)
    begin; /* region */
      in = values(innv,k);
      >>inv&(i)&(l)&(k)'n=
      if (in <> 0) then
        >>&(in)*bruttoin/100,
      else
        >>0,
    end;
  end;
end;
>>
end;
```

```
/* nsysys.src - nye likninger for sysselsatte (syss1) */  
/* Oversatt til Potroll av Rune Johansen. Sist oppdatert 5.7.96 */  
  
/* sysselsatte etter bosted (region). Netto pendling inkludert. */  
/* Ekstrafylket fordelt. */
```

```
addfun main;  
procedure main()  
begin;  
  
  >>addeq bottom  
  
  for(k=1;k<=7;k=k+1)  
  begin; /* region */  
    >>syss&(k)'n=  
    for(i=1;i<=2;i=i+1)  
    begin; /* kjonn */  
      for(j=1;j<=3;j=j+1)  
      begin; /* utdanning */  
        ii = convert(i); jj = convert(j);  
        koeff = "ekstra" || (ii) || (jj);  
        ekstra = getdata(koeff);  
        x = values(ekstra,k);  
        >>l&(i)&(j)&(k)-np&(i)&(j)&(k)  
        if (x <> 0) then  
          >>+&(x)*l&(i)&(j)8+  
        else  
          >>+  
        end;  
      end;  
    end;  
  end;  
  >>0,  
end;  
  
  >>;  
end;
```

```

/* nytilbud.src - tilbud av arbeid uten flytting */
/* Oversatt til Potroll av Rune Johansen. Sist oppdatert 25.09.96 */

/* koeffisient */
/* YPIND - regionale forskjeller i yrkesprosenter */

/* variable */
/* AS - endogen arbeidsstyrke etter kjonn, alder, utdanning, region */
/* B - befolkning etter kjonn, alder, utdanning og region */
/* JAS - endogen skaleringsvariabel. Skal vaere med aa sikre at total */
/* arbeidsstyrke matcher med tallet hentet fra Modag */

addfun main;
procedure main()
begin;

  >>addsym endogenous jas;

  >>addeq bottom

  for(i=1;i<=2;i=i+1)
  begin; /* kjonn */
    for(l=3;l<=9;l=l+1)
    begin; /* aldersgruppe 3-9 (16-74) */
      for(j=1;j<=3;j=j+1)
      begin; /* utdanning */
        ii = convert(i); ll = convert(l); jj = convert(j);
        koeff = "ypind" || (ii) || (ll) || (jj);
        ypind = getdata(koeff);
        for(k=1;k<=7;k=k+1)
        begin; /* region */
          y = values(ypind,k);
          >>as&(i)&(l)&(j)&(k)'n= jas*(
          >>if (b&(i)&(l)&(j)&(k)*&(y)*ypr&(i)&(l)&(j)+
            >>s.as&(i)&(l)&(j)&(k)
          >>lt b&(i)&(l)&(j)&(k) then
          >>b&(i)&(l)&(j)&(k)*&(y)*ypr&(i)&(l)&(j)+
          >>s.as&(i)&(l)&(j)&(k)
          >>else b&(i)&(l)&(j)&(k)
          >>),
        end;
      end;
    end;
  end;

  /* arbeidsstyrken summert skal vaere lik den totale verdien */
  /* hentet fra Modag */
  for(i=1;i<=2;i=i+1)
  begin; /* kjonn */
    for(l=3;l<=9;l=l+1)
    begin; /* aldersgruppe 3-9 (16-74) */
      for(j=1;j<=3;j=j+1)
      begin; /* utdanning */
        for(k=1;k<=7;k=k+1)
        begin; /* region */
          >>as&(i)&(l)&(j)&(k)+
        end;
      end;
    end;
  end;
  >>0=antot,

  >>;
end;

```


Macroer for demografiske oppdateringer i
REGARD

```
/* dgraf.src - Styringsmacroen for demografidelen i REGARD. */
/* Utfører kall paa macroene ninv, flytt, newborn, innbef, rb og utbef. */
/* Henter inn befolkningstall (innbef), og splitter opp demografiske */
/* storrelser til ettaarige aldersgrupper. Justerer befolkningsmatrisene */
/* med nettoinnvandring, flytting, fødsel og død (ninv, flytt, newborn og rb). */
/* Deretter oppdateres befolkningsvariablene som brukes i neste simulering */
/* (ARBEID), og resultatene eksporteres til resultatdatabasene (utbef). */
/* Oversatt til POTroll av Rune Johansen, sist oppdatert 17.2.96 */
```

```
addfun main;
procedure main()
begin;
  get number aar0;
  get number fra;

  /* Netto innvandring og flytting beregnes i modellen. Derfor */
  /* oppdateres ikke demografidata med disse første år */
  if (aar0 <> fra) then
  begin;

    /* henter den rendemografiske fremskrivningen fra utgangen av */
    /* foregående år, for deretter å korrigere denne for netto */
    /* innvandring i løpet av foregående år */
    &ninv; >>&(aar0)

    /* korrigerer fremskrivningen ovenfor innenlandske flyttinger, */
    /* og beregner dermed fremskrivninger for inngangen av året */
    &flytt; >>&(aar0)
  end; else
  begin;
    if (fra<=1996) then
    begin;
      /* Vi skal erstatte fremskrevne befolkninger med historiske */
      /* data for årene til og med 1996 */
      >>dofile ut2_bef&(fra) = ut4_bef&(fra);
    end;
    if (fra<=1995) then
    begin;
      /* for befolkning etter utdanning har vi bare tall t.o.m. 1995 */
      for(j=1;j<=3;j=j+1)
      begin;
        >>dofile ut2_bef&(j)&(fra) = ut4_bef&(j)&(fra);
      end;
    end;
  end;
end;

/* oppdaterer ettaarig befolkning ved inngangen til året med */
/* nyfødte og fremskriver befolkningen med kjennetegn ved utgangen */
/* av året denne fremskrevne befolkningen (BB) brukes i macroen */
/* innbef til å beregne eksogen befolkning til ARBEID */
&newborn; >>&(aar0)

/* aggregerer ettaarige fremskrivninger til REGARDS fleraarige */
/* aldersgrupper, og lagrer passende variable for senere bruk */
/* i modellen ARBEID. Bruker befolkningen ved inngangen av året */
/* men med kjennetegn som ved utgangen */
&innbef; >>&(aar0)

/* henter befolkningen ved inngangen av året, og lager en */
/* rendemografisk fremskrivning for utgangen av året, inkludert */
/* dødelighet, i motsetning til newborn. Den fremskrevne befolkningen */
/* BHAT brukes i utbef til å lage eksogene variable BUTG til modellen */
/* ARBEID */
&rb; >>&(aar0)

/* henter den rendemografiske fremskrivningen BHAT fra RB og lager */
/* variable BUTG til resten av modellen ARBEID for utgangen av året. */
&utbef; >>&(aar0)
```

end;

```

/* ninv.src - fremskriver inn- og utvandring for befolkningene i */
/* REGARDS regioner. Kalles av dgraf */
/* Oversatt til POTroll av Rune Johansen, sist oppdatert 17.2.97 */

addfun main;
procedure main()
begin;
  /* simuleringsaaret hentes inn */
  get number aar0;
  aar1 = aar0 - 1;
  aar2 = convert(aar1)||"A";

  /* forst hentes fremskrivningen over utdanning fra ARBEID, BUTG, og */
  /* fordeler denne over ettaarige alder i regionale kolonner */

  /* Den endogene variabelen BUTG indekstert over kjonn, alder, */
  /* utdanning og region fra ARBEID slaas sammen over kjonn og alder */
  /* til de lokale vektorene KHAT indekstert over utdanning og region. */
  /* Disse er 14x1 vektorer (7 aldersgrupper, 2 kjonn) */
  for(k=1;k<=7;k=k+1)
  begin; /* region */
    for(j=1;j<=3;j=j+1)
    begin; /* utdanning */
      for(i=1;i<=2;i=i+1)
      begin; /* kjonn */
        >>do khat&(i)&(j)&(k) = combine(
          for(l=3;l<=9;l=l+1)
          begin; /* alder */
            if (l<9) then
              >>values(ut1_butg&(i)&(l)&(j)&(k),&(aar2)),
            else
              >>values(ut1_butg&(i)&(l)&(j)&(k),&(aar2))
          end;
        >>);
      end;
    >>dofile ut2_khat&(j)&(k) = combine(khat1&(j)&(k), khat2&(j)&(k));
  end;
end;

/* nedbryting over ettaarig alder */
/* bruker her hjelpematriksen CU, som er en 118 X 14 matrise bestaaende */
/* av enere og nuller, og som beskriver overgangen mellom REGARDS */
/* fleraarige aldersgrupper, og ettaarige aldersgrupper. */

/* lokke over utdanningskategori */
for(j=1;j<=3;j=j+1)
begin;
  >>do x=ut2_bef&(j)&(aar1);
  >>do y=ut2_gu&(j);
  for(k=1;k<=7;k=k+1)
  begin;
    >>dofile ut2_bef&(j)&(aar1)c&(k) = submat(x,,col(x)==&(k));
    >>dofile ut2_gu&(j)c&(k) = submat(y,,col(y)==&(k));
  end;
  >>do ones=crmat(1,14,1);
  /* lokke over region */
  for(k=1;k<=7;k=k+1)
  begin;
    /* Lager nye fordelingstall mellom fleraarig og ettaarig */
    /* struktur for befolkning etter utdanning. Benytter */
    /* eksogene befolkningstall etter utdanning BEF(j,aar) */
    >>do x1=ut2_bef&(j)&(aar1)c&(k);

    /* aggregerer til fleraarig alder (til 14X1 vektorer) */
    >>do fler&(j)&(k) = matmult(transp(cu),x1);
    >>do ones2 = crmat(118,1,1);
  end;
end;

```

```

/* lager 118X14 matrise med den fleraarige aldersstrukturen */
/* i hver rad */
>>do fler = matmult(ones2,transp(fler&(j)&(k)));

/* NHU inneholder den ettaarige eksogene befolkn. etter utd. */
/* 'fordelt' til den tilhoerende fleraarige gruppen, og ordnet */
/* som en 118 X 14 matrise */
>>do nhu&(j)&(k) = cu*matmult(x1,ones);

/* beregner nye fordelingstall (andeler) for oppsplitting */
>>do nevner = if fler<>0 then fler else 1;
>>do ggu&(j)&(k) = nhu&(j)&(k)/nevner;

/* beregner ny ettaarig befolkning etter utdanning, ved */
/* aa splitte opp den endogent beregnede befolkning fra */
/* ARBEID med hjelp av den nylif beregnede oppsplittings- */
/* matrisen GGU */
>>dofile ut2_bhat&(j)&(k)=matmult(ggu&(j)&(k),ut2_khat&(j)&(k));
end;
end;

/* lager ettaarige bef.tall for hver utdanningskategori, ved aa */
/* aggregere over region */
for(j=1;j<=3;j=j+1)
begin;
  >>dofile ut2_bhat&(j)=partcomb(1,7
  for(k=1;k<=7;k=k+1)
  begin;
    >>,ut2_bhat&(j)&(k)
  end;
  >>);
end;

/* avstemming mot den rendemografiske fremskrivningen */

/* tar ut de indeksene i ettaarig alder som inneholder yrkesaktive */
/* menn og kvinner (16-74 aar). */
>>do amk = combine(seq(75,17),seq(161,103));

/* aggregerer over utdanning */
>>do buhat = ut2_bhat1 + ut2_bhat2 + ut2_bhat3;

/* lager ny utdanningfordeling, som tar utgangspunkt i den */
/* rendemografiske 172 X 7 befolkningsmatrisen BHAT(aar-1) for */
/* aaret for, og som regner riktig andel. */
for(j=1;j<=3;j=j+1)
begin;
  >>dofile ut2_bhat&(j) =
  >>(ut2_bhat&(j)/buhat)*submat(ut2_bhat&(aar1), amk);
end;

/* bhat&(j) er naa ettaarige fremskrivninger av utdanning som */
/* er konsistente med den rendemografiske fremskrivningen */

/* inn- og utvandring etter ettaarig alder slaas sammen til */
/* kolonnevektorer, en for hver region. */
for(k=1;k<=7;k=k+1)
begin;
  for(i=1;i<=2;i=i+1)
  begin;
    >>do inv&(i)&(k) = values(ut1_inv&(i)1&(k),&(aar2));
    >>do utv&(i)&(k) = values(ut1_utv&(i)1&(k),&(aar2));
    for(l=1;l<=8;l=l+1)
    begin;
      l2 = l + 1;
      >>do inv&(i)&(k) = addrow(inv&(i)&(k),&(l),
      >>values(ut1_inv&(i)&(l2)&(k),&(aar2)));
      >>do utv&(i)&(k) = addrow(utv&(i)&(k),&(l),
      >>values(ut1_utv&(i)&(l2)&(k),&(aar2)));
    end;
  end;
end;

```

```

        end;
        >>do inv&(i)&(k) = combine(inv&(i)&(k),0,0);
        >>do utv&(i)&(k) = combine(utv&(i)&(k),0,0);
    end;
    >>do utv&(k) = combine(utv1&(k), utv2&(k));
    >>do inv&(k) = combine(inv1&(k), inv2&(k));
end;

/* kolonnevektorene settes sammen i matriser */
>>do invx = inv1;
>>do utv = utv1;
for(i=1;i<=6;i=i+1)
begin;
    j=i+1;
    >>do invx = addcol(invx, &(i), inv&(j));
    >>do utv = addcol(utv, &(i), utv&(j));
end;
/* IN VX, UTV inneholder naa hhv inn- og utvandring beregnet i */
/* ARBEID slaatt sammen over kjonn, alder (11 grupper), og region */
/* I aldersgruppe 10 og 11 (gamle) har vi satt inn nuller, da vi */
/* antar ingen inn/utvandring blant disse. IN VX og UTV er derfor */
/* 22 X 7 matriser. Beregner saa */
/* netto innvandring i henhold til aldersgrupperingen i REGARD */
>>dofile ut2_ninv = invx - utv;

/* fordelingen av nettoinnvandringen paa ettaarig alder */
/* bruker her hjelpematriksen CC, som er en 172 X 22 matrise som */
/* beskriver aggregeringen fra alle REGARDS 11 aldersgrupper */
/* til ettaarig alder, helt analogt med CU */
/* Splitter forst opp den rendemografiske fremskrivningen BHAT(aar1) */
/* og nettoinnvandringen NINV i sju 172 X 1 kolonnevektorer (en for */
/* hver region */
>>do bh=ut2_bhat&(aar1);
for(k=1;k<=7;k=k+1)
begin;
    >>dofile ut2_bhat&(aar1)c&(k)=submat(bh,,col(bh)==&(k));
end;

/* Splitter opp netto innvandring i kolonner */
>>do x=ut2_ninv;
for(k=1;k<=7;k=k+1)
begin;
    >>dofile ut2_ninvc&(k)=submat(x,,col(x)==&(k));
end;

/* konstruerer hjelpematriksene BG1,...BG7 ved hjelp av den */
/* rendemografiske fremskrivningen og matrisen CC */
/* BG(k) er en 172 X 22 matrise, og bestaar av 22 kopier av kolonne */
/* nr k i den rendemografiske fremskrivningen satt ved siden av */
/* hverandre. */
for(k=1;k<=7;k=k+1)
begin;
    >>do bg&(k) = ut2_bhat&(aar1)c&(k);
    for(j=1;j<=21;j=j+1)
    begin;
        >>do bg&(k) = addcol(bg&(k), &(j), ut2_bhat&(aar1)c&(k));
    end;
    >>do bg&(k) = cc*bg&(k);

    /* Aggregerer rendemografisk fremskrivning fra ettaarig til */
    /* fleraarig alder. */
    >>do khat&(k) = matmult(transp(cc), ut2_bhat&(aar1)c&(k));

    /* Paa bakgrunn av de aggregerte tallene beregnes andelstall */
    /* for oppsplitting av nettoinnvandring fra fleraarig til */
    /* ettaarig alder. Vi antar med andre ord at den ettaarige */
    /* aldersstrukturen i nettoinnvandringen samsvarer med */
    /* aldersstrukturen i befolkningen i tilflyttingsregionen */
    >>do ghat&(k) = 1/khat&(k);

```

```
>>do g&(k) = matmult(bg&(k), ghat&(k));

/* 172 X 22 matrisen GG(k) inneholder andelstallene for */
/* oppsplittingen til ettaarig alder */
>>do gg&(k) = g&(k);
for(j=1;j<=21;j=j+1)
begin;
  >>do gg&(k) = addcol(gg&(k), &(j), g&(k));
end;
>>do gg&(k) = gg&(k)*cc;

/* Oppsplittingen av nettoinnvandringen til ettaarig alder */
/* foretas ved aa multiplisere 172 X 22 matrisen GG, med de */
/* fleraarige tallene for nettoinnvandring. */
>>do ut2_ninv&(k) = matmult(gg&(k), ut2_ninvc&(k));
end;

/* tallene legges i 172 X 7 matrisen NINV(aar) bestaaende av */
/* regionale kolonner med ettaarig aldersstruktur */
>>dofile ut2_ninv&(aar0) = ut2_ninv1;
for(i=1;i<=6;i=i+1)
begin;
  j=i+1;
  >>dofile ut2_ninv&(aar0) =
  >>addcol(ut2_ninv&(aar0),&(i),ut2_ninv&(j));
end;

/* tilslutt kan den ett-aarige rendemografiske fremskrivningen */
/* korrigeres for nettoinnvandring */
>>dofile ut2_nhat&(aar0) = ut2_bhat&(aar1) + ut2_ninv&(aar0);

/* netto innvandring etter utdanning. Tar her bare med de */
/* aldersgruppene (16-74 aar) som er yrkesaktive. Bruker */
/* derfor hjelpevariabelen AMK nok engang, for aa plukke ut */
/* de riktige elementene */
>>do buhat = ut2_bhat1 + ut2_bhat2 + ut2_bhat3;
for(j=1;j<=3;j=j+1)
begin;
  >>dofile ut2_ninvu&(j) =
  >>(ut2_bhat&(j)/buhat)*submat(ut2_ninv&(aar0),amk);
  >>dofile ut2_nhat&(j) = ut2_bhat&(j) + ut2_ninvu&(j);
end;

end;
```

```
end;
```

```

/* flytt.src - denne makroen justerer de ettaarige befolkningsmatrisene */
/* for netto innenlandske flyttinger ved aa bruke flytteresultatene for */
/* fleraarige kohortgrupper fra selve modellsystemet (delmod. ARBEID) */
/* Oversatt til POTroll av Rune Johansen, sist oppdatert 17.2.97 */

addfun main;
procedure main()
begin;
  /* simuleringsaaret leses inn */
  get number aar;
  aar1 = aar-1;
  aar2 = convert(aar1)||"A";

  /* forst beregnes flytting etter utdanningskategori. Henter */
  /* flyttetall beregnet i ARBEID, som settes sammen i matriser. */
  /* lokke over region */
  for(k=1;k<=7;k=k+1)
  begin;
    /* lokke over kjonn */
    for(i=1;i<=2;i=i+1)
    begin;
      /* lokke over utdanning */
      for(j=1;j<=3;j=j+1)
      begin;
        >>do infe&(i)&(j)&(k) = values(ut1_in&(i)3&(j)&(k),&(aar2));
        >>do utfe&(i)&(j)&(k) = values(ut1_ut&(i)3&(j)&(k),&(aar2));
        /* flyttetallene legges i kolonner etter alder. */
        for(l=1;l<=6;l=l+1)
        begin;
          l2=l+3;
          >>do infe&(i)&(j)&(k) = addrow(infe&(i)&(j)&(k),&(l),
            >>values(ut1_in&(i)&(l2)&(j)&(k),&(aar2)));
          >>do utfe&(i)&(j)&(k) = addrow(utfe&(i)&(j)&(k),&(l),
            >>values(ut1_ut&(i)&(l2)&(j)&(k),&(aar2)));
        end;
      end;
    end;
  end;
  /* nettoflytting etter utdanning */
  /* INFE(j,k) og UTFE(j,k) inneholder tallene for henholdsvis */
  /* innflytting og utflytting for utdanning j og region k. */
  /* Tallene er organisert som vektorer etter kjonn og alder, */
  /* aldersgruppe 3-9 (16-74 aar). Disse vektorene inneholder altsaa */
  /* 2*7=14 elementer. Differansen mellom disse vektorene, netto */
  /* flytting, lagres i vektorene NFKE(j,k). */
  /* Aldersgruppe 3-9 tilsvareer 59 ettaarige aldergrupper (16-74 aar) */
  /* og oppsplittingen av tallene hentet fra ARBEID, skjer ved aa */
  /* multiplisere med 118 X 14 matrisen FV. De ettaarige tallene */
  /* lagres i variabelen NFB */
  for(j=1;j<=3;j=j+1)
  begin;
    >>do infe&(j)&(k) = combine(infe1&(j)&(k),infe2&(j)&(k));
    >>do utfe&(j)&(k) = combine(utfe1&(j)&(k),utfe2&(j)&(k));
    >>do nfke&(j)&(k) = infe&(j)&(k)-utf&(j)&(k);
    /* fordeling av nettoflytting etter utdanning paa ettaarig */
    /* alder. aldersfordelingen er eksogen og def. gj. kolonnev. */
    /* fv&(utd.kategori) */
    >>dofile ut2_nfb&(j)&(k) = matmult(ut2_fv&(j), nfke&(j)&(k));
  end;
end;

/* kolonnevektorer for hver landsdel settes sammen til 118 X 7 */
/* matrisen NFB(j,aar-1) */
for(j=1;j<=3;j=j+1)
begin;
  /* midlertidige netto flyttetall */
  >>dofile ut2_nfb&(j)&(aar1) = ut2_nfb&(j)1;
  for(k=1;k<=6;k=k+1)
  begin;

```



```

    k2=k+1;
    >>dofile ut2_nfb&(j)&(aar1) =
    >>addcol(ut2_nfb&(j)&(aar1),&(k),ut2_nfb&(j)&(k2));
end;

/* foretar korrigerer av flyttetallene dersom den totale netto */
/* flyttingen er forskjellig fra null. */
>>do ones1 = crmat(7,1,1);
/* totale netto flyttetall for hver ettaarig aldersgruppe etter */
/* utdanning (118 X 1 vektor) (hvert element skal være lik null) */
>>do nfl = matmult(ut2_nfb&(j)&(aar1),ones1)/7;
>>do ones2 = crmat(1,7,1);
/* 118 X 7 matrise som inneholder korreksjonstallene */
>>do nfl = matmult(nfl,ones2);
/* endelig korrigererte netto flyttetall */
>>dofile ut2_nfb&(j)&(aar1) = ut2_nfb&(j)&(aar1) - nfl;

/* endelig fremskrivning av befolkning etter utdanning ved inngangen */
/* av aaret ved aa legge til netto flytting (etter utdanning) */
/* For aar<=1995 har vi historiske tall */
if (aar <= 1995) then
    >>dofile ut2_bef&(j)&(aar) = ut4_bef&(j)&(aar);
else
    >>dofile ut2_bef&(j)&(aar) = ut2_nhat&(j) + ut2_nfb&(j)&(aar1);

end;

/* totale flyttinger blant barn(B), voksne(V), og gamle(G) etter */
/* ettaarig alder */
/* I) Voksnes flyttinger */
>>do nfbv = ut2_nfb1&(aar1) + ut2_nfb2&(aar1) + ut2_nfb3&(aar1);
>>do x=nfbv;
/* elementene som inneholder tall for hhv menn og kvinner */
/* splitter opp */
>>do menn=seq(59);
>>do kvinner=seq(118,60);
/* lokke over region */
for(k=1;k<=7;k=k+1)
begin;
    >>do nfbvc&(k)=submat(x,,col(x)==&(k));
    >>do nfbv1&(k) = submat(nfbvc&(k), menn );
    >>do nfbv2&(k) = submat(nfbvc&(k), kvinner );
end;

/* II) Barnas flyttinger */
/* henter tall beregnet i ARBEID */
>>do ones=crmat(1,2,1);
for(k=1;k<=7;k=k+1)
begin;
    for(i=1;i<=2;i=i+1)
    begin;
        /* aldersgruppene 1 og 2 (barn slaas sammen, og det beregnes */
        /* netto flytting for barn, som lagres i variabelen NFKB */
        >>do nfkb&(i)&(k) =
        >>combine(values(ut1_inb&(i)1&(k),&(aar2)),
            >>values(ut1_inb&(i)2&(k),&(aar2))) -
        >>combine(values(ut1_utb&(i)1&(k),&(aar2)),
            >>values(ut1_utb&(i)2&(k),&(aar2)));
        /* fordeling til ettaarig alder */
        /* Forst lages fordelingsmatrisen FFB, ved aa multiplisere */
        /* (elementvis) fordelingstallene i 16 X 1 vektoren FLB (duplisert */
        /* til en 16 X 2 matrise, med 16 X 2 hjelpematrissen CBARN. */
        /* deretter foretas selve oppsplittingen ved aa matrise- */
        /* multiplisere 16 X 2 fordelingsmatrisen FFB med 2 X 1 vektoren */
        /* NFKB som inneholder netto flyttetall for barn etter REGARDS */
        /* aldersgrupper (aldersgruppe 1-2). Resultatet er altsaa en */
        /* 16 X 1 vektor som inneholder netto flyttetall for barn etter */
        /* ettaarig alder, og som lagres i vektoren NFBB */
        >>do ffb&(i) = cbarn*(matmult(ut2_flb&(i),ones));
    end;
end;

```

```

    >>do nfb&(i)&(k) = matmult(ffb&(i),nfkb&(i)&(k));
end;
end;

/* korrigerer barns flyttinger slik at total netto flytting for hver */
/* ettaarig aldersgruppe er lik 0 */
for(i=1;i<=2;i=i+1)
begin;
  /* midlertidige netto flyttetall barn */
  >>do flyttb&(i) = nfb&(i)1;
  for(k=1;k<=6;k=k+1)
  begin;
    k2=k+1;
    >>do flyttb&(i) = addcol(flyttb&(i),&(k),nfb&(i)&(k2));
  end;
  /* flyttb&(i) er 16 X 7 matriser med regionvise flyttetall barn */

  >>do ones1 = crmat(7,1,1);
  /* totale netto flyttetall for hver ettaarig aldersgruppe barn */
  /* (16 X 1 vektor) (hvert element skal være lik null) */
  >>do fb&(i) = matmult(flyttb&(i),ones1)/7;
  /* endelig korrigererte netto flyttetall barn */
  for(k=1;k<=7;k=k+1)
  begin;
    >>do nfb&(i)&(k) = nfb&(i)&(k) - fb&(i);
  end;
end;

/* III) Eldres flyttinger */
/* Setter inn nuller (ingen flytting blant eldre ??) */
>>do nfbg = crmat(11,1,0);
for(k=1;k<=7;k=k+1)
begin;
  for(i=1;i<=2;i=i+1)
  begin;
    /* Lager en 86 X 1 vektor som korresponderer med ettaarig */
    /* aldersgruppeinndeling. De 16 forste elementene er netto */
    /* flyttetall for barn (NFBB), de neste 59 er for yrkesaktive */
    /* 16-74 aar, mens de 11 siste er for eldre. Her settes det */
    /* inn nuller (NFBG). */
    >>do nffb&(i)&(k) = combine(nfb&(i)&(k), nfbv&(i)&(k),nfbg);
  end;
  /* slaar sammen over kjonn i 172 X 1 vektor */
  >>do nffb&(k)&(aar1) = combine(nffb1&(k),nffb2&(k));
end;

/* kolonner for hver region settes sammen til en 172 X 7 matrise */
/* for netto flytting, NFB */
>>dofile ut2_nfb&(aar1) = nffb1&(aar1);
for(k=1;k<=6;k=k+1)
begin;
  k2=k+1;
  >>dofile ut2_nfb&(aar1)=addcol(ut2_nfb&(aar1),&(k),nffb&(k2)&(aar1));
end;

/* endelig ettaarig fremskrivning korrigerert for netto innvandring */
/* og netto innenlandsk innflytting (legger til NFB) */
/* For aar<=1996 har vi historiske tall */
if (aar<=1996) then
  >>dofile ut2_bef&(aar) = ut4_bef&(aar);
else
  >>dofile ut2_bef&(aar) = ut2_nhat&(aar) + ut2_nfb&(aar1);

/* skalerer befolkning etter utdanning, slik at den alltid er lik */
/* den totale befolkningen naar vi summerer over utdanningskategori */
/* tar forst ut de yrkesaktive (16-74 aar) */
>>do amk = combine(seq(75,17),seq(161,103));
>>do yrke = submat(ut2_bef&(aar), amk);

```

```
>>do befut = ut2_bef1&(aar)+ut2_bef2&(aar)+ut2_bef3&(aar);  
for(j=1;j<=3;j=j+1)  
begin;  
  >>dofile ut2_bef&(j)&(aar) = (ut2_bef&(j)&(aar)/befut)*yrke;  
end;  
  
end;
```

```

/* newborn.src - Legger paa fødselstall som er generert ved aa gange */
/* fødselsrater med den kvinnelige delen av befolkningen ved inngangen */
/* av aaret. Brukes til aa fremskrive befolkningen slik at den har */
/* alderskjennetegn som ved utgangen av aaret, til senere bruk i macroen */
/* innbef */
/* befolkning etter utdanning aldres ogsaa her */
/* Oversatt til POfroll av Rune Johansen, sist oppdatert 17.2.97 */

addfun main;
procedure main()
begin;
  get number aar;

  /* FBM og FBK er 86 X 7 matriser som inneholder fødselsrater for hhv */
  /* menn og kvinner (regionvis). Siden det bare er kvinner i alder */
  /* 15-44 aar som antas aa føde barn, er ratene for andre aldersgrupper */
  /* lik null. vektoren BK inneholder de aktuelle indeksene */
  /* variablene FK&(k) og FM&(k) inneholder antall nyfødte hhv jenter */
  /* og gutter i region k */
  >>do bm=seq(44,15);
  >>do bk=seq(130,101);
  for(k=1;k<=7;k=k+1)
  begin;
    >>do fbmc&(k)=submat(ut2_fbm,bm,col(ut2_fbm)==&(k));
    >>do fbkc&(k)=submat(ut2_fbk,bm,col(ut2_fbk)==&(k));
    >>do fm&(k) =
    >>matmult(transp(fbmc&(k)),submat(ut2_bef&(aar),bk,&(k)));
    >>do fk&(k) =
    >>matmult(transp(fbkc&(k)),submat(ut2_bef&(aar),bk,&(k)));
  end;

  /* De regionvise tallene slaas sammen til 1 X 7 radvektorer som */
  /* inneholder antall nyfødte menn og kvinner */
  >>do fm=partcomb(1,7,fm1,fm2,fm3,fm4,fm5,fm6,fm7);
  >>do fk=partcomb(1,7,fk1,fk2,fk3,fk4,fk5,fk6,fk7);

  /* henter befolkningstallene ved utgangen av aaret foer (eventuelt */
  /* inkludert innvandring og flytting for aaret foer) */
  /* bruker historiske tall t.o.m 1995 */
  if (aar <= 1996) then
    >>dofile ut2_bb&(aar) = ut4_bef&(aar);
  else
    >>dofile ut2_bb&(aar) = ut2_bef&(aar);

  /* befolkningsmatrisen BB&(aar) oppdateres med tallene for nyfødte */
  /* menn i rad 1, kvinner i rad 88. Forskyver kohortene */
  >>dofile ut2_bb&(aar) = addrow(ut2_bb&(aar),0,fm);
  >>dofile ut2_bb&(aar) = addrow(ut2_bb&(aar),87,fk);

  /* fremskrevne eldre */
  >>do ut2_ga1 = submat(ut2_bb&(aar), 86) + submat(ut2_bb&(aar), 87);
  >>do ut2_ga2 = submat(ut2_bb&(aar), 173) + submat(ut2_bb&(aar), 174);

  >>do rad=transp(seq(7));
  /* legger inn de fremskrevne eldre i rad 86 og 173 i bb */
  >>dofile ut2_b1 = setrep(ut2_bb&(aar), ut2_ga1, 86, rad);
  >>dofile ut2_b2 = setrep(ut2_b1, ut2_ga2, 173, rad);
  >>dofile ut2_bb&(aar) = ut2_b2;

  /* fjerner de overflodige linjene */
  >>dofile ut2_bb&(aar) = delrow(ut2_bb&(aar), 87);
  >>dofile ut2_bb&(aar) = delrow(ut2_bb&(aar), 173);

  /* aldrer ogsaa befolkning etter utdanning */
  /* det er de som er 15-73 aar ved inngangen til aaret vi skal */
  /* ha med. De som er 74 aar (og blir 75 i lopet av aaret) skal */
  /* skyves ut */
  >>do befut1&(aar)=addrow(ut2_bef1&(aar),0,submat(ut2_bb&(aar),17));

```

```
>>do befut1&(aar)=addrow(befut1&(aar),60,submat(ut2_bb&(aar),103));
>>do befut2&(aar)=addrow(ut2_bef2&(aar),0,0);
>>do befut2&(aar)=addrow(befut2&(aar),60,0);
>>do befut3&(aar)=addrow(ut2_bef3&(aar),0,0);
>>do befut3&(aar)=addrow(befut3&(aar),60,0);
/* 75 aaringene skyves ut */
for(j=1;j<=3;j=j+1)
begin;
  >>do befut&(j)&(aar)=delrow(befut&(j)&(aar),60);
  >>dofile ut2_befut&(j)&(aar) = delrow(befut&(j)&(aar),119);
end;

/* bb&(aar) og befut&(j)&(aar) brukes i macroen innbef for aa beregne */
/* eksogene befolkningstall til modellen ARBEID */

end;
```

```

/* innbef.src - aggregerer befolkningen etter ettaarig alder for inngangen */
/* av aaret til REGARDS fleraarige aldersgruppeinndeling */
/* befolkningen ved inngangen av aaret (BB) er fremskrevet av newborn, slik */
/* at den har kjennetegn som ved utgangen */
/* Oversatt til POTroll av Rune Johansen, sist oppdatert 5.2.97 */

```

```
addfun main;
```

```
procedure main()
```

```
begin;
```

```
get number aar;
```

```
aar1 = convert(aar) || "A";
```

```
/* befolkningstall BB splittes opp i 172 X 1 kolonner etter region. */
```

```
/* Aggregerer til fleraarig inndeling med hjelp av matrisen CC */
```

```
/* Resultatene lagres i den lokale variabelen K(k,aar) */
```

```
>>do x=ut2_bb&(aar);
```

```
for(k=1;k<=7;k=k+1)
```

```
begin;
```

```
>>do bb&(aar)c&(k)=submat(x,,col(x)==&(k));
```

```
>>do k&(k)&(aar) = matmult(transp(cc), bb&(aar)c&(k));
```

```
end;
```

```
/* slaar sammen over region til 22 X 7 matrisen K(aar) */
```

```
>>do k&(aar)=partcomb(1,7
```

```
for(k=1;k<=7;k=k+1)
```

```
begin;
```

```
>>,k&(k)&(aar)
```

```
end;
```

```
>>);
```

```
/* oppsplitting over kjonn, menn i de 11 forste radene, kvinnene */
```

```
/* i de 11 siste */
```

```
>>do x=k&(aar);
```

```
>>do k1&(aar) = submat(x, row(x)<=11);
```

```
>>do k2&(aar) = submat(x, row(x)>11);
```

```
/* Gjør om de beregnede befolkningstallene til en tidsserie med */
```

```
/* aarlig periodisitet, lagrer i variabelen B indeksert over */
```

```
/* kjonn, alder og region. B brukes som eksogen variabel i ARBEID. */
```

```
for(k=1;k<=7;k=k+1)
```

```
begin;
```

```
for(i=1;i<=2;i=i+1)
```

```
begin;
```

```
for(l=3;l<=9;l=l+1)
```

```
begin;
```

```
>>do file ut1_b&(i)&(l)&(k)=overlay(
```

```
>>reshape(submat(k&(i)&(aar), &(l), &(k)), &(aar1)),
```

```
>>ut1_b&(i)&(l)&(k));
```

```
end;
```

```
end;
```

```
end;
```

```
/* fordeling etter utdanning ved inngangen av aaret */
```

```
/* Kun for yrkesaktive, aldersgruppe 3-9 (16-74 aar) */
```

```
/* Bruker 118 X 14 matrisen CU for aa aggregere fra ettaarig */
```

```
/* til fleraarig alder */
```

```
for(j=1;j<=3;j=j+1)
```

```
begin;
```

```
>>do x=ut2_befut&(j)&(aar);
```

```
for(k=1;k<=7;k=k+1)
```

```
begin;
```

```
>>do befut&(j)&(aar)c&(k)=submat(x,,col(x)==&(k));
```

```
>>do ku&(j)&(k) = matmult(transp(cu), befut&(j)&(aar)c&(k));
```

```
>>do y = ku&(j)&(k);
```

```
>>do ku1&(j)&(k) = submat(y, row(y)<=7 ,col(y)>=1);
```

```
>>do ku2&(j)&(k) = submat(y, row(y)>7 ,col(y)>=1);
```

```
end;
```

```
end;
```

```

/* til bruk i REGARD. B er eksogen variabel til neste runde i */
/* modellen ARBEID. */
for(k=1;k<=7;k=k+1)
begin;
  for(i=1;i<=2;i=i+1)
  begin;
    for(l=3;l<=9;l=l+1)
    begin;
      l2 = l - 2;
      for(j=1;j<=3;j=j+1)
      begin;
        >>dofile ut1_b&(i)&(l)&(j)&(k) =
        >>overlay(
        >>reshape(submat(ku&(i)&(j)&(k), &(l2)), &(aar1)),
        >>ut1_b&(i)&(l)&(j)&(k));
      end;
    end;
  end;
end;

/* spesial losning for aldersgruppe 3 med hoyere utdanning */
for(i=1;i<=2;i=i+1)
begin;
  >>dofile ut1_bm&(i)33 =
  for(k=1;k<=7;k=k+1)
  begin;
    >>ut1_b&(i)33&(k) +
  end;
  >>0;
end;

/* barn ved inngangen til aaret */
>>do barn1 = seq(16);
>>do barn2 = seq(102,87);
for(k=1;k<=7;k=k+1)
begin;
  for(i=1;i<=2;i=i+1)
  begin;
    >>do barn&(i)&(k) = submat(ut2_bb&(aar), barn&(i), &(k));
    >>do barn&(i)&(k) = matmult(transp(cbarn), barn&(i)&(k));
    for(l=1;l<=2;l=l+1)
    begin;
      >>dofile ut1_barn&(i)&(l)&(k) = overlay(
      >>reshape(submat(barn&(i)&(k), &(l)), &(aar1)),
      >>ut1_barn&(i)&(l)&(k));
    end;
  end;
end;

/* Naa er befolkningen lest inn, aggregert til REGARDS fleraarige */
/* aldersgrupper, og dermed klar til aa brukes som eksogen input */
/* i modellen ARBEID. */
end;

```

```
/* rb.src - foretar beregninger av dodelighet med utgangspunkt i matrisene */
/* for befolkningen etter ettaarig alder ved inngangen av aaret. */
/* Resultatet er en forelopig (rendemografisk) fremskrivning av */
/* befolkningen ved utgangen av aaret, foer nettoinnvandring og flytting */
/* Oversatt til POTroll av Rune Johansen, sist oppdatert 5.2.97 */

addfun main;
procedure main()
begin;
  get number aar;

  /* OL bestaar av overlevelsesrater for menn (SM) og kvinner (SK), */
  /* som begge er 87 X 7 matriser. OL er altsaa en 174 X 7 matrise */
  /* BH inneholder midlertidig befolkningstall (172 X 7) */
  >>do ol&(aar) = partcomb(2,1,ut2_sm&(aar),ut2_sk&(aar));
  >>do ol&(aar) = delrow(ol&(aar),174);
  >>do ol&(aar) = delrow(ol&(aar),87);

  /* henter inn befolkningen ved inngangen til aaret */
  /* eventuelt aldret og inkludert innvandring/flytting for aaret foer */
  >>do bh&(aar) = ut2_bb&(aar);

  /* korrigerer for dodelighet ved hj. av matrisene med overlevelsesrater */
  >>do ut2_bh&(aar) = ol&(aar)*bh&(aar);

  /* forelopig rendemografisk fremskrivning */
  >>dofile ut2_bhat&(aar) = ut2_bh&(aar);
  /* bhat&(aar) brukes f.eks i ninv, flytt og andre demografimacroer */

end;
```



```

/* utbef.src - aggregerer den forelopige demografiske framskrivningen */
/* fra makroen RB til REGARDS fleraarige aldersgruppeinndeling */
/* Oversatt til POTroll av Rune Johansen, sist oppdatert 05.02.97 */

addfun main;
procedure main()
begin;
  get number aar;
  aar1 = convert(aar) || "A";

  /* aggregerer fra ettaarig til fleraarig med hjelp av matrisen CC */
  >>do x=ut2_bhat&(aar);
  for(k=1;k<=7;k=k+1)
  begin;
    >>do bhat&(aar)c&(k)=submat(x,,col(x)==&(k));
    >>do kh&(k)&(aar) = matmult(transp(cc), bhat&(aar)c&(k));
  end;
  /* slaar sammen over region til 22 X 7 matrise */
  >>do kh&(aar)=partcomb(1,7)
  for(k=1;k<=7;k=k+1)
  begin;
    >>,kh&(k)&(aar)
  end;
  >>);

  /* oppsplitting over kjonn, menn i de forste 11, kvinner de siste 11 */
  >>do km = seq(11);
  >>do kk = seq(22,12);
  >>do kh1&(aar) = submat(kh&(aar), km);
  >>do kh2&(aar) = submat(kh&(aar), kk);

  /* lager nye verdier for den eksogene variabelen BUTG i ARBEID */
  /* Denne er indeksert over kjonn, alder, og region, og maa ikke */
  /* forveksles med den endogene versjonen, som ogsaa er indeksert */
  /* over utdanning */
  for(k=1;k<=7;k=k+1)
  begin; /* region */
    for(i=1;i<=2;i=i+1)
    begin; /* kjonn */
      for(l=1;l<=9;l=l+1)
      begin; /* alder */
        >>dofile ut1_butg&(i)&(l)&(k)=overlay(
          >>reshape(submat(kh&(i)&(aar), &(l), &(k)), &(aar1)),
          >>ut1_butg&(i)&(l)&(k));
      end;
    end;
  end;
end;
end;
end;

```

Macroer for eksport av resultater i
REGARD

```
/* export.src - Macro for aa styre eksportering av resultater fra */
/* REGARD (befolkning, arbeidsstyrke, sysselsetting og markedsleie). */
/* Oversatt av Rune Johansen. Sist oppdatert 31. juli 1996 */

addfun main;
procedure main()
begin;
  get navn; /* navn paa kjoringen */
  get number aar1; /* forste aar */
  get number aar2; /* siste aar */

  navn = lower(navn);

  /* befolkningstall */
  print("befolkning");
  &exbef; >>&(navn) &(aar1) &(aar2)

  /* arbeidsstyrke */
  print("arbeidsstyrke");
  &arbst; >>&(navn) &(aar1) &(aar2)

  /* Sysselsetting */
  print("sysselsetting");
  &sysjr; >>&(navn) &(aar1) &(aar2)

  /* Markedsleie */
  print("markedsleie");
  &xml; >>&(navn) &(aar1) &(aar2)

end;
```

```

/* exbef.src - eksporterer sammensatte matriser for befolkning og */
/* nettoflytting etter ettaarig alder. En fil gir totaltall, mens */
/* de tre andre gir resultatene for alder 16-74 aar brutt ned over */
/* utdanningskategorier */
/* Oversatt av Rune Johansen. Sist oppdatert 5.2.1997 */

addfun main;
procedure main()
begin;
  get navn; /* navn paa kjoringen */
  get number aar1; /* forste aar */
  get number aar2; /* siste aar */

  aar3 = aar2+1;
  antaar = aar2 - aar1 + 1;
  ant2 = antaar + 1;

  navn = lower(navn);

  /* Behandler forst totaltall over alle aldersgrupper 0-85 aar */

  /* lager forst kolonnene som inneholder indekser for kjonn og alder */
  >>do k1 = combine(crmat(86,1,1),crmat(86,1,2));
  >>do k2 = combine(seq(86),seq(86))-1;

  /* setter k1 og k2 sammen */
  >>do kj = partcomb(1,2,k1,k2);

  /* aarstallene legges inn */
  for(i=aar1;i<=aar2;i=i+1)
  begin;
    >>do ar=crmat(172,1,&(i));
    /* befolkningstall */
    >>do b&(i) = partcomb(1,3,ar,kj,ut2_bef&(i));
    /* flytting */
    >>do n&(i) = partcomb(1,3,ar,kj,ut2_nfb&(i));
  end;
  >>do ar=crmat(172,1,&(aar3));
  >>do b&(aar3) = partcomb(1,3,ar,kj,ut2_bef&(aar3));

  /* setter sammen matrisene for alle aarene til en stor matrise */
  >>dofile beftot = partcomb(&(ant2),1
  for(i=aar1;i<=aar3;i=i+1)
  begin;
    >>,b&(i)
  end;
  >>);

  >>dofile nfbtot = partcomb(&(antaar),1
  for(i=aar1;i<=aar2;i=i+1)
  begin;
    >>,n&(i)
  end;
  >>);

  /* Gjør saa samme operasjon for tall brutt ned over utdanning, */
  /* kun for alder 16-74 aar */

  /* lager forst kolonnene som inneholder indekser for kjonn og alder */
  >>do k1 = combine(crmat(59,1,1),crmat(59,1,2));
  >>do k2 = combine(seq(59),seq(59))+15;

  /* setter k1 og k2 sammen */
  >>do kj = partcomb(1,2,k1,k2);

  /* aarstallene legges inn */
  for(j=1;j<=3;j=j+1)
  begin;

```

```

    for(i=aar1;i<=aar2;i=i+1)
    begin;
        >>do ar=crmat(118,1,&(i));
        /* befolkningstall */
        >>do b&(j)&(i) = partcomb(1,3,ar,kj,ut2_bef&(j)&(i));
        /* netto flytting */
        >>do n&(j)&(i) = partcomb(1,3,ar,kj,ut2_nfb&(j)&(i));
    end;
    >>do ar=crmat(118,1,&(aar3));
    >>do b&(j)&(aar3) = partcomb(1,3,ar,kj,ut2_bef&(j)&(aar3));
end;

/* setter sammen matrisene for alle aarene og alle utdannings- */
/* kategoriene til en stor matrise */
for(j=1;j<=3;j=j+1)
begin;
    >>dofile befutd&(j) = partcomb(&(ant2),1
    for(i=aar1;i<=aar3;i=i+1)
    begin;
        >>,b&(j)&(i)
    end;
    >>);

    >>dofile nfbutd&(j) = partcomb(&(antaar),1
    for(i=aar1;i<=aar2;i=i+1)
    begin;
        >>,n&(j)&(i)
    end;
    >>);
end;

/* alle matrisene beftot, nfbtot, befutd1,2,3 og nfbutd1,2,3 er laget */
/* maa saa sende de ut av Troll */
print("sender matriser ut av troll");

/* eksporterer beftot */
for(i=1;i<=10;i=i+1)
begin;
    >>do beftotc&(i) = submat(beftot,,col(beftot)==&(i));
end;
>>do hfoopen("beftot.txt","w");
>>do fp1 = hfogetfp("beftot.txt");
>>do hfprintf(fp1,"%5.0f %2.0f %3.0f %7.0f %7.0f %7.0f %7.0f %7.0f %7.0f %7.0f
>>\n",
>>beftotc1,beftotc2,beftotc3,beftotc4,beftotc5,
>>beftotc6,beftotc7,beftotc8,beftotc9,beftotc10);
>>do hfclose(fp1);

/* eksporterer nfbtot */
for(i=1;i<=10;i=i+1)
begin;
    >>do nfbtotc&(i) = submat(nfbtot,,col(nfbtot)==&(i));
end;
>>do hfoopen("nfbtot.txt","w");
>>do fp2 = hfogetfp("nfbtot.txt");
>>do hfprintf(fp2,"%5.0f %2.0f %3.0f %7.0f %7.0f %7.0f %7.0f %7.0f %7.0f %7.0f
>>\n",
>>nfbtotc1,nfbtotc2,nfbtotc3,nfbtotc4,nfbtotc5,
>>nfbtotc6,nfbtotc7,nfbtotc8,nfbtotc9,nfbtotc10);
>>do hfclose(fp2);

/* eksporterer befutd og nfbutd */
for(j=1;j<=3;j=j+1)
begin;
    for(i=1;i<=10;i=i+1)
    begin;
        >>do befutd&(j)c&(i) = submat(befutd&(j),,
        >>col(befutd&(j))==&(i));
    end;
end;

```

```
>>do h fopen("befutd&(j).txt", "w");
>>do fp3 = hfogetfp("befutd&(j).txt");
>>do hfprintf(fp3, "%5.0f %2.0f %3.0f %7.0f %7.0f %7.0f %7.0f %7.0f %7.0f %7.0f
>>\n",
>>befutd&(j)c1, befutd&(j)c2, befutd&(j)c3, befutd&(j)c4, befutd&(j)c5,
>>befutd&(j)c6, befutd&(j)c7, befutd&(j)c8, befutd&(j)c9, befutd&(j)c10);
>>do hfclose(fp3);

for(i=1; i<=10; i=i+1)
begin;
    >>do nfbutd&(j)c&(i) = submat(nfbutd&(j), ,
    >>col(nfbutd&(j))==&(i));
end;
>>do h fopen("nfbutd&(j).txt", "w");
>>do fp4 = hfogetfp("nfbutd&(j).txt");
>>do hfprintf(fp4, "%5.0f %2.0f %3.0f %7.0f %7.0f %7.0f %7.0f %7.0f %7.0f %7.0f
>>\n",
>>nfbutd&(j)c1, nfbutd&(j)c2, nfbutd&(j)c3, nfbutd&(j)c4, nfbutd&(j)c5,
>>nfbutd&(j)c6, nfbutd&(j)c7, nfbutd&(j)c8, nfbutd&(j)c9, nfbutd&(j)c10);
>>do hfclose(fp4);
end;

end;
```

```

/* arbst.src - Macro som lager tre filer for arbeidsstyrke etter */
/* utdanning og alder i henhold til REGARDS fleraarige aldersinndeling */
/* Oversatt av Rune Johansen. Sist oppdatert 5.2.1997 */

addfun main;
procedure main()
begin;
  get navn; /* navn paa kjoringen */
  get number aar1; /* forste aar */
  get number aar2; /* siste aar */
  aar0 = aar1 - 1;
  aar3 = aar2 + 1;
  antaar = aar2 - aar1 + 1;
  ant2 = antaar + 1;

  navn = lower(navn);

  /* forst hentes arbeidsstyrken beregnet av ARBEID inn */
  /* og settes sammen til matriser over kjonn, utdanning og aar */
  for(i=1;i<=2;i=i+1)
  begin; /* kjonn */
    for(j=1;j<=3;j=j+1)
    begin; /* utdanning */
      for(a=aar0;a<=aar2;a=a+1)
      begin; /* aarstall */
        year = convert(a) || "A";
        for(k=1;k<=7;k=k+1)
        begin;
          >>do xas&(i)&(j)&(k)=values(ut1_as&(i)3&(j)&(k), &(year));
        end;
        >>do as&(i)&(j)&(a)=transp(combine(
          for(k=1;k<=7;k=k+1)
          begin; /* region */
            /* begynner med aldersgruppe 3 (16-74 aar) */
            if (k<7) then
              >>xas&(i)&(j)&(k),
            else
              >>xas&(i)&(j)&(k)
            end;
          >>));
        for(l=4;l<=9;l=l+1)
        begin; /* aldersgruppe 4-9 (20-74 aar) */
          l2 = l-3;
          for(k=1;k<=7;k=k+1)
          begin;
            >>do xas&(i)&(l)&(j)&(k)=
              >>values(ut1_as&(i)&(l)&(j)&(k), &(year));
            end;
            >>do x=transp(combine(
              for(k=1;k<=7;k=k+1)
              begin; /* region */
                if (k<7) then
                  >>xas&(i)&(l)&(j)&(k),
                else
                  >>xas&(i)&(l)&(j)&(k)
                end;
              >>));
            >>do y=as&(i)&(j)&(a);
            >>do as&(i)&(j)&(a) =
              >>addrow(y,&(l2),x);
          end;
        end;
      end;
    end;
  end;
end;

/* lager saa vektorer med andre personkjennetegn som skal inngaa i */
/* resultatmatrisen, bl.a. kjonn, og alder */

```

```

>>do me=crmat(7,1,1);
>>do kv=crmat(7,1,2);
/* REGARDS aldersgrupper for yrkesaktive (16-74 aar) */
>>do alder = combine(3,4,5,6,7,8,9);

/* setter personkjennetegnene og tilhørende tall for arbeidsstyrken */
/* sammen til store matriser, en for hver utdanningskategori */
for(j=1;j<=3;j=j+1)
begin;
  for(a=aar0;a<=aar2;a=a+1)
  begin;
    >>do aar = crmat(7,1,&(a));
    >>do as&(j)&(a) = partcomb(2,4,aar,aar,me,kv,alder,alder,
                              >>as1&(j)&(a),as2&(j)&(a));
  end;
  >>do file arbsty&(j)=partcomb(&(ant2),1
  for(a=aar0;a<=aar2;a=a+1)
  begin;
    >>,as&(j)&(a)
  end;
  >>);
end;

/* matrisene arbsty1,2,3 er naa laget, og eksporteres */

for(j=1;j<=3;j=j+1)
begin;
  for(i=1;i<=10;i=i+1)
  begin;
    >>do arbsty&(j)c&(i) = submat(arbsty&(j),,col(arbsty&(j))==&(i));
  end;
  >>do hfopen("arbst&(j).txt","w");
  >>do fp1 = hfogetfp("arbst&(j).txt");
>>do hfprintf(fp1,"%5.0f %2.0f %3.0f %7.0f %7.0f %7.0f %7.0f %7.0f %7.0f %7.0f
>>\n",
  >>arbsty&(j)c1,arbsty&(j)c2,arbsty&(j)c3,arbsty&(j)c4,arbsty&(j)c5,
  >>arbsty&(j)c6,arbsty&(j)c7,arbsty&(j)c8,arbsty&(j)c9,arbsty&(j)c10);
  >>do hfclose(fp1);
end;

end;

```



```
/* sysjr.src - Macro som eksporterer tall for sysselsetting for alle */
/* naeringer og alle fremskrivningsaar. */
/* Oversatt av Rune Johansen. Sist oppdatert 27.1.1997 */

addfun main;
procedure main()
begin;
  get navn; /* navn paa kjoringen */
  get number aar1; /* forste aar */
  get number aar2; /* siste aar */
  aar0 = aar1 - 1;
  aar3 = aar2 + 1;
  antaar = aar2 - aar1 + 1;
  ant2 = antaar + 1;

  navn = lower(navn);

  /* henter inn sektorlista */
  sektor = getdata("modag");
  ns = nvals(sektor);

  for(a=aar0;a<=aar2;a=a+1)
  begin;
    year = convert(a)||"A";
    se = values(sektor,1);
    sel=se;
    if (se=="92S") then
      sel=921;
    if (se=="93S") then
      sel=931;
    if (se=="93K") then
      sel=932;
    if (se=="94S") then
      sel=941;
    if (se=="94K") then
      sel=942;
    if (se=="95S") then
      sel=951;
    if (se=="95K") then
      sel=952;
    for(k=1;k<=8;k=k+1)
    begin;
      >>do xls&(se)&(k)=values(ut1_ls&(se)&(k),&(year));
    end;
    >>do ls&(a)=transp(combine(&(a),&(sel),
    for(k=1;k<=8;k=k+1)
    begin;
      if (k<8) then
        >>xls&(se)&(k),
      else
        >>xls&(se)&(k)
    end;
  >>));
  for(m=2;m<=ns;m=m+1)
  begin;
    se = values(sektor,m);
    sel=se;
    if (se=="92S") then
      sel=921;
    if (se=="93S") then
      sel=931;
    if (se=="93K") then
      sel=932;
    if (se=="94S") then
      sel=941;
    if (se=="94K") then
      sel=942;
    if (se=="95S") then
```

```
        sel=951;
        if (se=="95K") then
            sel=952;
        for(k=1;k<=8;k=k+1)
        begin;
            >>do xls&(se)&(k)=values(ut1_ls&(se)&(k),&(year));
        end;
        >>do x=transp(combine(&(a),&(sel),
        for(k=1;k<=8;k=k+1)
        begin;
            if (k<8) then
                >>xls&(se)&(k),
            else
                >>xls&(se)&(k)
            end;
        >>));
        >>do ls&(a)=addrow(ls&(a),&(m)-1,x);
    end;
end;
>>dofile ljr = partcomb(&(ant2),1
for(a=aar0;a<=aar2;a=a+1)
begin;
    >>,ls&(a)
end;
>>);

/* matrisen ljr med sysselsettingstall er naa laget, */
/* og maa eksporteres */

for(i=1;i<=10;i=i+1)
begin;
    >>do ljrc&(i) = submat(ljr,,col(ljr)==&(i));
end;
>>do hfoopen("sysjr.txt","w");
>>do fp1=hfogetfp("sysjr.txt");
>>do hfprintf(fp1,"%5.0f %4g %7.0f %7.0f %7.0f %7.0f %7.0f %7.0f %7.0f %7.0f
>>\n",
    >>ljrc1,ljrc2,ljrc3,ljrc4,ljrc5,ljrc6,ljrc7,ljrc8,ljrc9,ljrc10);
>>do hfclose(fp1);

end;
```

```

/* xml.src - Macro som eksporterer tall for markedsleie */
/* Oversatt av Rune Johansen. Sist oppdatert 6.2.1997 */

addfun main;
procedure main()
begin;
  get navn; /* navn paa kjoringen */
  get number aar1; /* forste aar */
  get number aar2; /* siste aar */
  aar0 = aar1 - 1;
  aar3 = aar2 + 1;
  antaar = aar2 - aar1 + 1;
  ant2 = antaar + 1;

  navn = lower(navn);

  for(a=aar0;a<=aar2;a=a+1)
  begin;
    year = convert(a)||"A";
    for(i=1;i<=2;i=i+1)
    begin; /* lokke over kjonn */
      for(k=1;k<=7;k=k+1)
      begin; /* region */
        >>do xmlhttp&(i)l&(k)=values(ut1_xml&(i)l&(k),&(year));
      end;
      >>do xml&(a)&(i)=transp(combine(&(a),&(i),1,
      for(k=1;k<=7;k=k+1)
      begin; /* region */
        if (k<7) then
          >>xml&(i)l&(k),
        else
          >>xml&(i)l&(k)
        end;
      >>));
      for(j=2;j<=3;j=j+1)
      begin;
        for(k=1;k<=7;k=k+1)
        begin;
          >>do xmlhttp&(i)&(j)&(k)=values(ut1_xml&(i)&(j)&(k),&(year));
        end;
        >>do x=transp(combine(&(a),&(i),&(j),
        for(k=1;k<=7;k=k+1)
        begin;
          if (k<7) then
            >>xml&(i)&(j)&(k),
          else
            >>xml&(i)&(j)&(k)
          end;
        >>));
        >>do xml&(a)&(i)=addrow(xml&(a)&(i),&(j)-1,x);
      end;
    end;
    >>do xml&(a) = partcomb(2,1,xml&(a)1,xml&(a)2);
  end;
  /* slaar alt sammen i en stor matrise */
  >>dofile xml = partcomb(&(ant2),1
  for(a=aar0;a<=aar2;a=a+1)
  begin;
    >>,xml&(a)
  end;
  >>);

  /* matrisen xml med tall for markedsleie er naa laget, */
  /* og maa eksportereres */

  for(i=1;i<=10;i=i+1)
  begin;
    >>do xmlc&(i) = submat(xml,,col(xml)==&(i));
  end;

```

```
end;
>>do hfopen("xml.txt", "w");
>>do fp1=hfogetfp("xml.txt");
>>do hfprintf(fp1, "%5.0f %2.0f %3.0f %7.5f %7.5f %7.5f %7.5f %7.5f %7.5f %7.5f
>>\n",
>>xmlc1,xmlc2,xmlc3,xmlc4,xmlc5,xmlc6,xmlc7,xmlc8,xmlc9,xmlc10);
>>do hfclose(fp1);

end;
```

Macroer som genererer likninger i
den ettårige versjonen av
REGARD

NB!!

Den ettårige versjonen av REGARD er i skrivende stund (5.12.97) bare delvis ferdig. Vi har kun laget macroene som genererer likningene. Her har vi bare tatt med de macroene som er *forskjellige* fra de som inngår i den fleråerige (dagens) versjon av REGARD. For å kunne lage en fullstendig ettårig versjon av REGARD, må man lage koeffisienter som inngår i likningene (og som brukes av macroene her til å lage likningene. Dessuten må datagrunnlaget konverteres til ettårig struktur.

```
/* arbeid.src - macro for aa generere likninger i REGARD */
/* ettaarig versjon */
/* Programmert av Rune Johansen. Sist oppdatert 5.12.1997 */

addfun main;
procedure main()
begin;

  /* aksesserer og søker paa databasen med koeffisienter */
  >>access koef type trolldb id ./koeff.trl mode r;
  >>search data koef;

  >>usemod arbeid;modedit;

  /* likninger for sammensetningen av sysselsetning etter kjonn */
  /* og utdanning */
  &nysyssutd;

  /* etterspørsel etter arbeid. Lager ogsaa likninger for syss for */
  /* ekstrasfylket */
  &ettersp2;

  /* sysselsetningen for hver region er den samme enten man splitter */
  /* opp etter sektor eller kjonn/utdanning. */
  &gam2;

  /* fremskrivning av personer etter utdanning paa grl.nivaa */
  &nyutd;

  /* flyttelikninger */
  /* flytting etter personkjennetegn */
  &flyttmod;
  /* generelle flyttelikninger */
  &flytmod2;
  /* flytting av barn
  &flytmod5;

  /* inn- og utvandring */
  &innutv;

  /* nye likninger for sysselsatte */
  &nysyss;

  /* Likningene for yrkesprosenter, tidligere laget ved et kall paa */
  /* macroen yprmod, utgaar fordi yrkesprosenten naa blir beregnet */
  /* eksogent ved hjelp av MOSART-tall (i formodellen). */

  /* tilbud av arbeid (uten flytting) */
  &nytilbud;

  /* Rendemografiske fremskrivninger */
  &ninv; /* netto innvandring */
  &flytt; /* flytting */
  &newborn; /* aldring og nyfodte */
  &rb; /* dodelighet */

  /* De genererte likningene lagres paa modellen ARBEID */
  >>filemod;

end;
```

```

/* nyutd.src - fremskrivning av personer etter utdanning paa gr.nivaa */
/* Oversatt til Potroll av Rune Johansen. Sist oppdatert 4.7.96 */

/* koeffisienter */
/* UTIND - indekser for vekst i utdanning */
/* UTFORD - fordeling av personer med hoyere utdanning (3) */
/*      i ald.gr 3 (16-19) */

/* variable */
/* B - Folketall etter kjonn, alder, utdanning og region */
/* BUTG (1) - endogent folketall etter kjonn, alder, utdanning og region */
/*      ved utgangen av aaret */
/* BUTG (2) - Folketall etter kjonn, alder og region ved utg. av aaret */
/*      (aggregert over utdanningskategorier). Eksogen variabel */
/*      beregnet i demografidelen. */
/* M - vekstrater for personer i aldersgrupper fra Mosart */
/* JB - endogen justering av utdanningsfordeling */
/* BM - absolutt tilveskt i aldersgruppe 3 hele landet */

/* Den endogent beregnede variabelen BUTG, beregnet av likningene generert */
/* av denne macroen, blir ved utgangen av aaret justert i samsvar med den */
/* eksogene BUTG beregnet i demografidelen (UTBEF). Denne forelopige */
/* eksogene variabelen omfatter ikke inn- og utvandring samt flytting. */

addfun main;
procedure main()
begin;

    >>addeg bottom

    /* befolkning etter kjonn, alder, utdanning og region */
    /* lokke over kjonn */
    for(i=1;i<=2;i=i+1)
    begin;
        /* lokke over utdanningskategori */
        for(j=1;j<=3;j=j+1)
        begin;
            /* aldersgrupper 20-74 aar */
            for(l=20;l<=74;l=l+1)
            begin;
                ii = convert(i); jj = convert(j); ll = convert(l);
                koeff = "utind" || (ii) || (ll) || (jj);
                utind = getdata(koeff); /* indekser vekst utdanning */
                for(k=1;k<=7;k=k+1)
                begin; /* region */
                    ut = values(utind,k);
                    >>butg&(i)&(l)&(j)&(k)'n=
                    if (ut <> 0) then
                    begin;
                        >>&(ut)*m&(i)&(l)&(j)*jb&(i)&(l)&(k)'n*
                        >>b&(i)&(l)&(j)&(k)-
                    end;
                    /* tallene justeres ved aa trekke i fra netto */
                    /* innvandring, fordi MOSART-tallene for befolknings- */
                    /* vekst inkluderer denne, mens REGARD behandler denne */
                    /* separat */
                    >>inv&(i)&(l)&(k)'n+utv&(i)&(l)&(k)'n,
                end;
            end;
        end;
    end;

    /* de fleste i aldersgruppe 16-19 har IKKE hoyere utdanning */
    for(j=1;j<=2;j=j+1)
    begin;
        for(l=16;l<=19;l=l+1)
        begin;
            ii = convert(i); jj = convert(j); ll = convert(l);
            koeff = "utind" || (ii) || (ll) || (jj);
            utind = getdata(koeff);
            /* lokke over region */

```



```
    for(k=1;k<=7;k=k+1)
    begin;
        ut = values(utind,k);
        >>butg&(i)&(l)&(j)&(k) 'n=
        if (ut <> 0) then
        begin;
            >>&(ut) *m&(i)&(l)&(j) *jb&(i)&(l)&(k) 'n*b&(i)&(l)&(j)&(k)
            >>-inv&(i)&(l)&(k) 'n+utv&(i)&(l)&(k) 'n,
        end;
    end;
end;
end;
/* spesiallosning for de faa i ald.gr.16-19 som har hoyere utd. */
for(l=16;l<=19;l=l+1)
begin;
    ii = convert(i);ll = convert(l);
    koeff = "utford"||(ii)||(ll);
    utford = getdata(koeff);
    /* lokke over region */
    for(k=1;k<=7;k=k+1)
    begin;
        ut = values(utford,k);
        >>butg&(i)&(l)3&(k) 'n=
        if (ut <> 0) then
            >>&(ut) *bm&(i)&(l)3,
        else
            >>0,
        end;
    end;
end;
/* lokke over aldersgruppe 16-74 aar */
for(l=16;l<=74;l=l+1)
begin;
    /* lokke over region */
    for(k=1;k<=7;k=k+1)
    begin;
        /* utdanningskategori */
        for(j=1;j<=3;j=j+1)
        begin;
            if (j == 1) then
                >>butg&(i)&(l)&(j)&(k)
            else
                >>+butg&(i)&(l)&(j)&(k)
            end;
        end;
        >>=butg&(i)&(l)&(k),
    end;
end;
end;
end;
end;
end;
```

```

/* flyttmod.src - Flytting etter personkjennetegn */
/* Oversatt til Potroll av Rune Johansen. Sist oppdatert 5.5.97 */

/* koeffisienter */
/* FLYT - flytterater for aldersgruppe 3 (16-19) hoy utdanning (3) */
/* FLP - estimerte parametre i flyttemodellen */

/* variable */
/* UT - utflyttere etter kjonn, alder, utdanning, fraflyttingsregion */
/* UTB - utflyttede barn */
/* IN - innflyttere til regionen etter kjonn, alder, utdanning */
/* INB - innflyttede barn */
/* B - folketall */
/* S.UTB - eksogen korreksjon barn, utflytting */
/* S.INB - eksogen korreksjon barn, innflytting */
/* S.UT - eksogen tilpasning basisaar utflytting inkl barn */
/* S.IN - eksogen tilpasning basisaar innflytting inkl barn */
/* XML - markedsleie */

```

```

addfun main;
procedure main()
begin;

  >>addeq bottom

  /* beregning av flytting for personer med utdanningsfordeling */

  /* aldersgruppe 16-19 og utdanning 3 (hoyere), spesialbehandling */
  for(i=1;i<=2;i=i+1)
  begin; /* kjonn */
    for(l=16;l<=19;l=l+1)
    begin;
      for(k=1;k<=7;k=k+1)
      begin; /* fra- og tilflyttingsregion */
        ii = convert(i); kk = convert(k); ll = convert(l);
        koeff = "flyt" || (ii) || (ll) || "3" || (kk);
        flytt = getdata(koeff);
        >>ut&(i)&(l)3&(k)'n=(
          for(k1=1;k1<=7;k1=k1+1)
          begin; /* tilflyttingsregion */
            fl = values(flytt,k1);
            >>&(fl)+
          end;
          >>0)*b&(i)&(l)3&(k)/1000+s.ut&(i)&(l)3&(k),
          >>in&(i)&(l)3&(k)'n=
          for(k1=1;k1<=7;k1=k1+1)
          begin; /* fraflyttingsregion */
            fl = values(flytt,k1);
            if (fl <> 0) then
              >>&(fl)*b&(i)&(l)3&(k1)/1000+
            end;
          >>s.in&(i)&(l)3&(k),
        end;
      end;
    end;
  end;

  /* generelt tilfelle, aldersgruppe 45-74 */

  for(i=1;i<=2;i=i+1)
  begin; /* kjonn */
    for(k=1;k<=7;k=k+1)
    begin; /* fraflyttingsregion */
      for(l=45;l<=74;l=l+1)
      begin; /* aldersgruppe */
        for(j=1;j<=3;j=j+1)
        begin; /* utdanning */
          ii = convert(i); jj = convert(j);
          kk = convert(k); ll = convert(l);

```

```

    koeff = "flyt" || (ii) || (ll) || (jj) || (kk);
    flytt = getdata(koeff);
    >>ut&(i)&(l)&(j)&(k) 'n=(
    for(k1=1;k1<=7;k1=k1+1)
    begin; /* fraflyttingregion */
        fl = values(flytt,k1);
        >>&(fl)+
    end;
    >>0)*b&(i)&(l)&(j)&(k)/1000+s.ut&(i)&(l)&(j)&(k),
    >>in&(i)&(l)&(j)&(k) 'n=
    for(k1=1;k1<=7;k1=k1+1)
    begin; /* fraflyttingsregion */
        fl = values(flytt,k1);
        if (fl <> 0) then
            >>&(fl)*b&(i)&(l)&(j)&(k1)/1000+
        end;
    >>s.in&(i)&(l)&(j)&(k),
    end;
end;
end;
end;

/* utflytting for andre enn ungdommer med hoyeste utdanning */
/* generelt tilfelle aldersgruppe 20-44 */

for(i=1;i<=2;i=i+1)
begin; /* kjonn */
    for(k=1;k<=7;k=k+1)
    begin; /* fraflyttingsregion */
        for(l=20;l<=44;l=l+1)
        begin; /* aldersgrupper */
            for(j=1;j<=3;j=j+1)
            begin; /* utdanning */
                ii = convert(i); jj = convert(j);
                kk = convert(k); ll = convert(l);
                koeff = "flp" || (ii) || (ll) || (jj) || (kk);
                flytt = getdata(koeff);
                >>ut&(i)&(l)&(j)&(k) 'n=b&(i)&(l)&(j)&(k)*((
                for(k1=1;k1<=6;k1=k1+1)
                begin; /* tilflyttingsregion */
                    fl = values(flytt,k1);
                    if (k1<k) then
                        if (fl <> 0) then
                            begin;
                                >>exp(&(fl)*xml&(i)&(j)&(k)/xml&(i)&(j)&(k1))+
                            end; else
                                >>1+
                            else
                                begin;
                                    k2=k1+1;
                                    if (fl<>0) then
                                        begin;
                                            >>exp(&(fl)*xml&(i)&(j)&(k)/xml&(i)&(j)&(k2))+
                                        end; else
                                            >>1+
                                        end;
                                end;
                            end;
                        end;
                    end;
                >>0)/(
                for(k1=1;k1<=6;k1=k1+1)
                begin; /* tilflyttingsregion */
                    fl = values(flytt,k1);
                    if (k1<k) then
                        if (fl<>0) then
                            begin;
                                >>exp(&(fl)*xml&(i)&(j)&(k)/xml&(i)&(j)&(k1))+
                            end; else
                                >>1+
                            else
                                begin;

```


end;

```

/* flytmod2.src - Likninger for innflytting til hver region. */
/* Oversatt til Potroll av Rune Johansen. Sist oppdatert 5.5.97 */

/* koeffisienter */
/* FLP - flyttematriser */

/* variable */
/* IN - innflytting etter kjonn, alder, utdanning for hver region */
/* B - folketall etter kjonn, alder, utdanning og fraflyttingsregion */
/* XML - markedsleie etter kjonn, utdanning og region */

/* NB!! Maa beregne flytteandeler paa nytt, da utflytting fra en gitt */
/* region til en annen i macroen flyttmod ikke blir beregnet spesifikt. */

addfun main;
procedure main()
begin;

  >>addeq bottom

  /* innflytting */

  for(i=1;i<=2;i=i+1)
  begin; /* kjonn */
    for(l=20;l<=44;l=l+1)
    begin; /* aldersgruppene 20-44 aar */
      for(j=1;j<=3;j=j+1)
      begin; /* utdanning */
        for(k=1;k<=7;k=k+1)
        begin; /*innflyttingsregion */
          >>in&(i)&(l)&(j)&(k)'n=
          for(k1=1;k1<=6;k1=k1+1)
          begin; /* fraflyttingsregion. Alle andre enn 1 */
            if (k1<k) then
            begin;
              k2=k1;k3=1;
            end; else
            begin;
              k2=k1+1;k3=0;
            end;
            ii = convert(i); ll = convert(l);
            jj = convert(j); mm = convert(k2);
            koeff = "flp" || (ii) || (ll) || (jj) || (mm);
            flytt = getdata(koeff);
            k4 = k - k3;
            fl = values(flytt,k4);
            >>b&(i)&(l)&(j)&(k2)
            /* beregner utflyttingsandel til region k */
            if (fl <> 0) then
              >>*exp(&(fl)*xml&(i)&(j)&(k2)/xml&(i)&(j)&(k))
            >>/(
            for(k5=1;k5<=6;k5=k5+1)
            begin;
              if (k5<k2) then
                k6=k5;
              else
                k6=k5+1;
              fl = values(flytt,k5);
              if (fl <> 0) then
                >>exp(&(fl)*xml&(i)&(j)&(k2)/xml&(i)&(j)&(k6))+
            end;
            >>1)+
          end;
        >>s.in&(i)&(l)&(j)&(k),
      end;
    end;
  end;

  /* behandling av aldersgruppe 16-19 aar UTEN hoyere utdanning */
  for(l=16;l<=19;l=l+1)

```

```

begin;
  for(j=1;j<=2;j=j+1)
    begin; /* utdanning */
      for(k=1;k<=7;k=k+1)
        begin; /* innflyttingsregion */
          >>in&(i)&(l)&(j)&(k)'n=
          for(k1=1;k1<=6;k1=k1+1)
            begin;
              if (k1<k) then
                begin;
                  k2=k1;k3=1;
                end; else
                begin;
                  k2=k1+1;k3=0;
                end;
              ii = convert(i); jj = convert(j);
              mm = convert(k2); ll= convert(l);
              koeff = "flp" || (ii) || (ll) || (jj) || (mm);
              flytt = getdata(koeff);
              k4 = k - k3;
              fl = values(flytt,k4);
              >>b&(i)&(l)&(j)&(k2)
              if (fl <> 0) then
                >>*exp(&(fl)*xml&(i)&(j)&(k2)/xml&(i)&(j)&(k))
              >>/ (
              for(k5=1;k5<=6;k5=k5+1)
                begin;
                  if (k5<k2) then
                    k6=k5;
                  else
                    k6=k5+1;
                  fl = values(flytt,k5);
                  if (fl <> 0) then
                    >>exp(&(fl)*xml&(i)&(j)&(k2)/xml&(i)&(j)&(k6))+
                end;
              >>1)+
            end;
          >>s.in&(i)&(l)&(j)&(k),
        end;
      end;
    end;
  end;
>>
end;

```

```

/* flytmod5.src - Likninger for flytting av barn */
/* Oversatt til Potroll av Rune Johansen. Sist oppdatert 5.5.97 */

/* koeffisienter */
/* BFLIN - flytterater barn */
/* FLP - flytterater voksne */

/* variable */
/* FLRB - Flytterate for barn fra region til region */

addfun main;
procedure main()
begin;

  >>addeq bottom

  for(i=1;i<=2;i=i+1)
  begin; /* kjonn */
    for(l=0;l<=15;l=l+1)
    begin; /* aldersgruppe (barn) */
      lv = l + 25; /* aldersgruppe 'moedregenerasjon' */
      for(k=1;k<=7;k=k+1)
      begin; /* fraflyttingsregion */
        ii = convert(i); ll = convert(l); kk = convert(k);
        koeff = "bflin" || (ii) || (ll) || (kk);
        bflin = getdata(koeff);
        for(k1=1;k1<=7;k1=k1+1)
        begin; /* tilflyttingsregion */
          if (k1 <> k) then
          begin;
            >>flrb&(i)&(l)&(k)&(k1)'n=
            bf = values(bflin,k1);
            if (k1<k) then
              k2=k1;
            else
              k2=k1-1;
            if (bf <> 0) then
            begin;
              >>&(bf)*
              for(j=1;j<=3;j=j+1)
              begin; /* utdanning voksne kvinner ('moedre') */
                ll = convert(lv); jj = convert(j);
                kk = convert(k);
                koeff = "flp2" || (ll) || (jj) || (kk);
                flp = getdata(koeff);
                fl = values(fl, k2);
                >>b2&(lv)&(j)&(k)
                if (k1<k) then
                begin;
                  if (fl <> 0) then
                    >>*exp(&(fl)*xml2&(j)&(k)/xml2&(j)&(k1))
                  else
                    >>1
                end; else
                begin;
                  if (fl <> 0) then
                    >>*exp(&(fl)*xml2&(j)&(k)/xml2&(j)&(k2))
                  else
                    >>1
                end;
              >>/
            for(k3=1;k3<=6;k3=k3+1)
            begin;
              fl = values(fl, k3);
              if (k3<k) then
              begin;
                if (fl <> 0) then
                  >>exp(&(fl)*xml2&(j)&(k)/xml2&(j)&(k3))+

```



```

        else
            >>1+
        end; else
        begin;
            k4 = k3+1;
            if (f1 <> 0) then
                >>exp(&(f1)*xml2&(j)&(k)/xml2&(j)&(k4))+
            else
                >>1+
            end;
        end;
        >>1)+
    end;
    >>0)/(
    for(j=1;j<=3;j=j+1)
    begin;
        >>b2&(lv)&(j)&(k)+
    end;
    >>0),
    end; else
    >>0,
        end; /* slutt test k1<>k */
    end; /* slutt k1-lokke */
end; /* slutt k-lokke */
/* relasjoner for utflytting for barn */
for(k=1;k<=7;k=k+1)
begin; /* fraflyttingsregion */
    >>utb&(i)&(l)&(k) 'n=barn&(i)&(l)&(k)*(
    for(k1=1;k1<=6;k1=k1+1)
    begin;
        if (k1<k) then
            begin;
                k2=k1;k3=1;
            end; else
            begin;
                k2=k1+1;k3=0;
            end;
            >>flrb&(i)&(l)&(k)&(k2)+
        end;
        >>0)+s.utb&(i)&(l)&(k),
    end;
end;
end;

/* innflytting barn */

for(i=1;i<=2;i=i+1)
begin; /* kjonn */
    for(l=0;l<=15;l=l+1)
    begin; /* aldersgruppe barn 0-15 aar */
        for(k=1;k<=7;k=k+1)
        begin; /* innflyttingsregion */
            >>inb&(i)&(l)&(k) 'n=
            for(k1=1;k1<=6;k1=k1+1)
            begin;
                if (k1<k) then
                    k2 = k1;
                else
                    k2 = k1 + 1;
                >>barn&(i)&(l)&(k2)*flrb&(i)&(l)&(k2)&(k)+
            end;
            >>s.inb&(i)&(l)&(k),
        end;
    end;
end;
end;
end;
>>
end;

```

```

/* innutv.src - Beregning av inn- og utvandring */
/* Oversatt til Potroll av Rune Johansen. Sist oppdatert 5.5.97 */

/* koeffisienter */
/* UTVAN - fordeling av utvandring paa kjonn og alder */
/* INVAN - fordeling av innvandring paa kjonn og alder */

/* Variable */
/* INV - innvandring etter kjonn alder og region */
/* UTV - utvandring etter kjonn alder og region */
/* B - befolkning etter kjonn alder utdanning og region */
/* BARN - barnetall */

addfun main;
procedure main()
begin;

  >>addeq bottom

  for(i=1;i<=2;i=i+1)
  begin; /* kjonn */
    for(l=16;l<=74;l=l+1)
    begin; /* aldersgruppe 16-74 aar */
      ii = convert(i); ll = convert(l);
      koeff = "utvan" || (ii) || (ll);
      utvand = getdata(koeff);
      for(k=1;k<=7;k=k+1)
      begin; /* region */
        ut = values(utvand,k);
        >>utv&(i)&(l)&(k)'n=
        /* aggregerer befolkning over utdanning */
        if (ut <> 0) then
          >>&(ut)*(b&(i)&(l)1&(k)+b&(i)&(l)2&(k)+b&(i)&(l)3&(k))/1000,
        else
          >>0,
        end;
      end;
    end;

  for(l=0;l<=15;l=l+1)
  begin; /* aldersgruppe 0-15 aar */
    ii = convert(i); ll = convert(l);
    koeff = "utvan" || (ii) || (ll);
    utvand = getdata(koeff);
    for(k=1;k<=7;k=k+1)
    begin;
      ut = values(utvand,k);
      >>utv&(i)&(l)&(k)'n=
      if (ut <> 0) then
        >>&(ut)*barn&(i)&(l)&(k)/1000,
      else
        >>0,
      end; /* slutt k-lokke */
    end;
  end;

  /* beregning av innvandring */

  >>bruttoin'n=
  for(i=1;i<=2;i=i+1)
  begin; /* kjonn */
    for(k=1;k<=7;k=k+1)
    begin; /* region */
      for(l=0;l<=74;l=l+1)
      begin; /* aldersgrupper 0-74 aar */
        >>utv&(i)&(l)&(k)+
      end;
    end;
  end;
  >>nettoin,

```

```
for(i=1;i<=2;i=i+1)
begin; /* kjonn */
  for(l=0;l<=74;l=l+1)
  begin; /* alder */
    ii = convert(i); ll = convert(l);
    koeff = "invan" || (ii) || (ll);
    innv = getdata(koeff);
    for(k=1;k<=7;k=k+1)
    begin; /* region */
      in = values(innv,k);
      >>inv&(i)&(l)&(k)'n=
      if (in <> 0) then
        >>&(in)*brutttoin/100,
      else
        >>0,
    end;
  end;
end;
>>
end;
```

```

/* nytilbud.src - tilbud av arbeid uten flytting */
/* Oversatt til Potroll av Rune Johansen. Sist oppdatert 5.5.97 */

/* koeffisient */
/* YPIND - regionale forskjeller i yrkesprosenter */

/* variable */
/* AS - endogen arbeidsstyrke etter kjonn, alder, utdanning, region */
/* B - befolkning etter kjonn, alder, utdanning og region */
/* JAS - endogen skaleringsvariabel. Skal vaere med aa sikre at total */
/* arbeidsstyrke matcher med tallet hentet fra Modag */

addfun main;
procedure main()
begin;

  >>addsym endogenous jas;

  >>addeg bottom

  for(i=1;i<=2;i=i+1)
  begin; /* kjonn */
    for(l=16;l<=74;l=l+1)
    begin; /* aldersgruppe 16-74 */
      for(j=1;j<=3;j=j+1)
      begin; /* utdanning */
        ii = convert(i); ll = convert(l); jj = convert(j);
        koeff = "ypind" || (ii) || (ll) || (jj);
        ypind = getdata(koeff);
        for(k=1;k<=7;k=k+1)
        begin; /* region */
          y = values(ypind,k);
          >>as&(i)&(l)&(j)&(k) 'n= jas*(
          >>if (b&(i)&(l)&(j)&(k)*&(y)*ypr&(i)&(l)&(j)+
              >>s.as&(i)&(l)&(j)&(k)
          >>lt b&(i)&(l)&(j)&(k)) then
          >>b&(i)&(l)&(j)&(k)*&(y)*ypr&(i)&(l)&(j)+
          >>s.as&(i)&(l)&(j)&(k)
          >>else b&(i)&(l)&(j)&(k)
          >>),
        end;
      end;
    end;
  end;

  /* arbeidsstyrken summert skal vaere lik den totale verdien */
  /* hentet fra Modag */
  for(i=1;i<=2;i=i+1)
  begin; /* kjonn */
    for(l=16;l<=74;l=l+1)
    begin; /* aldersgruppe 16-74 */
      for(j=1;j<=3;j=j+1)
      begin; /* utdanning */
        for(k=1;k<=7;k=k+1)
        begin; /* region */
          >>as&(i)&(l)&(j)&(k)+
        end;
      end;
    end;
  end;
  >>0=antot,

  >>;
end;

```

```

/* ninv.src - Lager likninger for aa fremskrive befolkningen med inn- og */
/* utvandring i REGARDS regioner. Ettaarige aldersgrupper */
/* Programmert av Rune Johansen, sist oppdatert 16.5.97 */

```

```

addfun main;
procedure main()
begin;

  >>addeq bottom
  for(k=1;k<=7;k=k+1)
  begin; /* region */
    for(i=1;i<=2;j=j+1)
    begin; /* kjonn */
      for(l=16;l<=74;l=l+1)
      begin; /* alder */
        >>ninv&(i)&(l)&(k)'n=inv&(i)&(l)&(k)(-1)-utv&(i)&(l)&(k)(-1),
        >>nhat&(i)&(l)&(k)'n=b&(i)&(l)&(k)(-1)+ninv&(i)&(l)&(k),
        >>buhat&(i)&(l)&(k)'n=
        >>butg&(i)&(l)1&(k)+butg&(i)&(l)2&(k)+butg&(i)&(l)3&(k),
        for(j=1;j<=3;j=j+1)
        begin; /* utdanning */
          >>ninv&(i)&(l)&(j)&(k)'n=
          >>(butg&(i)&(l)&(j)&(k)/buhat&(i)&(l)&(k))*ninv&(i)&(l)&(k),
          >>nhat&(i)&(l)&(j)&(k)'n=
          >>b&(i)&(l)&(j)&(k)(-1)+ninv&(i)&(l)&(j)&(k),
        end;
      end;
    end;
  end;
end;
>>;

end;

```

```

/* flytt.src - denne makroen lager likninger som justerer befolknings- */
/* tallene med netto innenlandske flyttinger */
/* Programmert av Rune Johansen, sist oppdatert 5.5.97 */

```

```

addfun main;
procedure main()
begin;

  >>addeg bottom
  for(i=1;i<=2;i=i+1)
  begin; /* kjonn */
    for(l=16;l<=74;l=l+1)
    begin; /* alder */
      for(j=1;j<=3;j=j+1)
      begin; /* utdanning */
        for(k=1;k<=7;k=k+1)
        begin; /* region */
          >>nfl&(i)&(l)&(j)&(k)'n=
          >>in&(i)&(l)&(j)&(k)(-1)-ut&(i)&(l)&(j)&(k)(-1),
          end;
          >>nfl&(i)&(l)&(j)'n=(
          for(k=1;k<=7;k=k+1)
          begin; /* region */
            nfl&(i)&(l)&(j)&(k)+
          end;
          >>0)/7,
          for(k=1;k<=7;k=k+1)
          begin;
            >>nfb&(i)&(l)&(j)&(k)'n=nfl&(i)&(l)&(j)&(k)-nfl&(i)&(l)&(j),
            >>bef&(i)&(l)&(j)&(k)'n=
            >>nhat&(i)&(l)&(j)&(k)+nfb&(i)&(l)&(j)&(k),
          end;
        end;
      end;
    end;
  end;

  /* totale flyttinger blant barn(B), voksne(V), og gamle(G) */
  /* I) Voksnes flyttinger */

  for(i=1;i<=2;i=i+1)
  begin; /* kjonn */
    for(l=16;l<=74;l=l+1)
    begin; /* alder */
      for(k=1;k<=7;k=k+1)
      begin; /* region */
        >>nfb&(i)&(l)&(k)'n=
        >>nfb&(i)&(l)1&(k)+nfb&(i)&(l)2&(k)+nfb&(i)&(l)3&(k),
        >>bef&(i)&(l)&(k)'n=nhat&(i)&(l)&(k)+nfb&(i)&(l)&(k),
      end;
    end;
  end;

  /* II) Barnas flyttinger */

  for(i=1;i<=2;i=i+1)
  begin; /* kjonn */
    for(l=0;l<=15;l=l+1)
    begin; /* alder */
      for(k=1;k<=7;k=k+1)
      begin; /* region */
        >>nflb&(i)&(l)&(k)'n=
        >>inb&(i)&(l)&(k)(-1)-utb&(i)&(l)&(k)(-1),
      end;
      >>nfl&(i)&(l)'n=(
      for(k=1;k<=7;k=k+1)
      begin; /* region */
        nflb&(i)&(l)&(k)+

```

```
end;
>>0)/7,
for(k=1;k<=7;k=k+1)
begin;
  >>nfb&(i)&(l)&(k)'n=nflb&(i)&(l)&(k)-nfl&(i)&(l),
  >>bef&(i)&(l)&(k)'n=bhat&(i)&(l)&(k)+nfb&(i)&(l)&(k),
end;
end;
end;

/* III) Eldres flyttinger */
for(k=1;k<=7;k=k+1)
begin;
  for(i=1;i<=2;i=i+1)
  begin;
    for(l=75;l<=85;l=l+1)
    begin;
      >>nfb&(i)&(l)&(k)'n=0,
      >>bef&(i)&(l)&(k)'n=bhat&(i)&(l)&(k)+nfb&(i)&(l)&(k),
    end;
  end;
end;
end;
>>
end;
```

```

/* newborn.src - Lager likninger for aa aldre befolkningen */
/* Inkluderer ny kohort med 0-aaringer */
/* Programmert av Rune Johansen, sist oppdatert 15.5.97 */

addfun main;
procedure main()
begin;

  >>addeg bottom
  /* aldring */
  for(i=1;i<=2;i=i+1)
  begin;
    for(k=1;k<=7;k=k+1)
    begin;
      >>bb&(i)85&(k)'n=bef&(i)85&(k)+bef&(i)84&(k),
      for(l=84;l>=1;l=l-1)
      begin;
        >>bb&(i)&(l)&(k)'n=bef&(i)&(l-1)&(k),
      end;
    end;
  end;

  /* nyfodte inkluderes */
  for(k=1;k<=7;k=k+1);
  begin;
    >>bb100&(k)'n=
    for(l=15;l<=44;l=l+1)
    begin;
      >>fbm&(l)&(k)*bb2&(l)&(k)+
    end;
    >>0,
    >>bb200&(k)'n=
    for(l=15;l<=44;l=l+1)
    begin;
      >>fbk&(l)&(k)*bb2&(l)&(k)+
    end;
    >>0,
  end;

  for(i=1;i<=2;i=i+1)
  begin;
    for(k=1;k<=7;k=k+1)
    begin;
      >>befut&(i)161&(k)'n=bef&(i)16&(k),
      >>befut&(i)162&(k)'n=0,
      >>befut&(i)163&(k)'n=0,
      for(l=17;l<=74;l=l+1)
      begin;
        for(j=1;j<=3;j=j+1)
        begin;
          >>befut&(i)&(l)&(j)&(k)'n=
          >>(butg&(i)&(l)&(j)&(k)/buhat&(i)&(l)&(k))*
          >>bb&(i)&(l)&(k),
        end;
      end;
    end;
  end;
  >>;

end;

```



```
/* rb.src - lager likninger for aa inkludere dodelighet */
/* Programmert av Rune Johansen, sist oppdatert 28.4.97 */

addfun main;
procedure main()
begin;

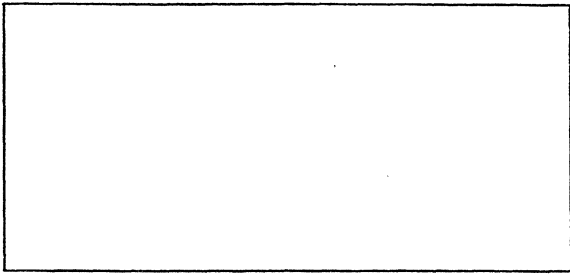
  for(i=1;i<=2;i=i+1)
  begin;
    for(l=0;l<=85;l=l+1)
    begin;
      for(k=1;k<=7;k=k+1)
      begin;
        >>b&(i)&(l)&(k)'n=bb&(i)&(l)&(k)*ol&(i)&(l)&(k),
        for(j=1;j<=3;j=j+1)
        begin;
          >>b&(i)&(l)&(j)&(k)'n=
          >>(butg&(i)&(l)&(j)&(k)/buhat&(i)&(l)&(k))*
          >>b&(i)&(l)&(k),
        end;
      end;
    end;
  end;
end;
>>;

end;
```

De sist utgitte publikasjonene i serien Notater fra Forskningsavdelingen

- 95/26 I. Aslaksen, T. Fagerli og H.A. Gravningsmyhr: Produksjon og konsum i husholdningene
- 95/29 B.E. Naug: Eksport- og importlikninger i KVARTS
- 95/31 B.E. Naug: Etterspørsel etter arbeidskraft - en litteraturoversikt
- 95/35 T.J. Klette: Vekst og produktivitet i norsk industri. Hovedrapport fra et NFR-prosjekt
- 95/40 L. Lerskau: Oversikt over konjunkturindikatorer i databasen NORMAP og FAME
- 95/46 B.E. Naug: Estimering av eksportrelasjoner på disaggregerte kvartalsdata
- 95/47 K. Moum: Beregning av bruttoproduksjon og eierinntekt i boligsektoren i nasjonalregnskapet - noen metodiske synspunkter
- 95/52 T. Kornstad: Simulering av konsum og arbeidstilbud i et livsløpsperspektiv
- 95/56 A. Langørgen: Faktorer bak kommunale variasjoner i utgifter til sosialhjelp og barnevern
- 95/58 T. W. Karlsen: Energimarkedet fra 1973 og fram mot 2010
- 96/3 I. M. Smestad: Valg under usikkerhet: En analyse av eksperimentdata basert på kvalitative valghandlingsmodeller
- 96/8 B. Lian og K. O. Aarbu: Dokumentasjon av LOTTE-AS
- 96/9 D. Fredriksen: Datagrunnlaget for modellen MOSART, 1993
- 96/10 S. Grepperud og A. C. Bøeng: Konsekvensene av økte oljeavgifter for råoljepris og etterspørsel etter olje. Analyser i PETRO og WOM
- 96/16 K. Gerdrup: Inntektsfordeling og økonomisk vekst i norske fylker: En empirisk studie basert på data for perioden 1967-93
- 96/31 A. Bruvoll og H. Wiig: Konsekvenser av ulike håndteringsmåter for avfall
- 96/33 M. Rolland: Militærutgifter i Norges prioriterte samarbeidsland
- 96/35 A.C. Hansen: Analyse av individers preferanser over lotterier basert på en stokastisk modell for usikre utfall
- 96/36 B.H. Vatne: En dynamisk spillmodell: Dokumentasjon av dataprogrammer
- 96/44 K.-G.Lindquist og B.E.Naug: Makroøkonometriske modeller og konkurranseevne.
- 96/45 R. Golombek og S. Kverndokk (red): Modeller for elektrisitets- og gassmarkedene i Norge, Norden og Europa.
- 96/53 F.R. Aune: Konsekvenser av en nordisk avgiftsharmonisering på elektrisitetsområdet.
- 97/2 E. Berg og K. Rypdal: Historisk utvikling og fremskrivning av forbruket av noen miljøskadelige produkter
- 97/5 Å. Cappelen: SSBs arbeid med investeringsrelasjoner: erfaringer og planer
- 97/30 K.-G. Lindquist: Database for energiintensive næringer. Tall fra industristatistikken
- 97/35 A. Langørgen: Faktorer bak variasjoner i kommunal ressursbruk til pleie og omsorg
- 97/36 S. E. Førre: Registerdataene i lys av industristatistikken
- 97/37 K. Gimming: Virkninger på prisutviklingen på naturgass i Vest-Europa ved innføring av felles karbonavgift
- 97/39 E.Holmøy og Ø.Thøgersen (red.): Virkninger av strukturpolitiske reformer: Forslag til konkrete forskningsprosjekter
- 97/41 E. Holmøy: En presisering av hva som skal menes med tilbudskurven for arbeid i en generell likevektsmodell
- 97/45 A. Katz, B.M. Larsen, K.S. Eriksen og T. Jensen: Transport og makroøkonomi – en samkjøring av GODMOD-3 og MSG-6
- 97/52 J. Nordøy: Nyttan av forventningsbaserte konjunkturindekser ved predikering av konsum
- 97/68 R. Johansen: Modell for regional analyse av arbeidsmarked og demografi. Teknisk dokumentasjon

Notater



Tillatelse nr.
159 000/502

B *Returadresse:*
Statistisk sentralbyrå
Postboks 8131 Dep.
N-0033 Oslo

Statistisk sentralbyrå

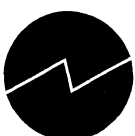
Oslo:
Postboks 8131 Dep.
0033 Oslo

Telefon: 22 86 45 00
Telefaks: 22 86 49 73

Kongsvinger:
Postboks 1260
2201 Kongsvinger

Telefon: 62 88 50 00
Telefaks: 62 88 50 30

ISSN 0806-3745



Statistisk sentralbyrå
Statistics Norway