

DET KGL. NORSKE VIDENSKABERS SELSKAB, MUSEET

rappoort

ZOOLOGISK SERIE 1977-4

ORNITOLOGG

En beskrivelse av et
programsystem for foredling og
informasjonsuttrekking av materiale
samlet inn med datalogger

Arne Venstad



Universitetet i Trondheim

K. norske Vidensk. Selsk. Mus. Rapport Zool. Ser. 1977-4

ORNITOLOGG

En beskrivelse av et programsystem for foredling
og informasjonsuttrekking av
materiale samlet inn med datalogger

av

Arne Venstad

Universitetet i Trondheim
Det Kgl. Norske Videnskabers Selskab, Museet

Trondheim, mars 1977

ISBN 82-7126-130-4

REFERAT

Venstad, Arne. 1977. ORNITOLOGG. En beskrivelse av et programsystem for foredling og informasjonsuttrekking av materiale samlet inn med datalogger. *K. norske Vidensk. Selsk. Mus. Rapport Zool. Ser. 1977-4.*

Somrene 1970, 1971, 1972 og 1973 ble det ved hjelp av datalogger registrert og samlet inn data vedrørende forskjellige fugler i rugeperioden. Det ble målt temperatur i og rundt egg under ruging samt faste klimaparametre som temperatur, nedbør, luftfuktighet og lys. De registrerte målingene ble først overført til papirhullbånd, senere til magnetbånd. De innsamlede data er bearbeidet og danner grunnlaget for et programsystem som trekker ut ønsket informasjon og gjør enkle statistiske beregninger. Det er lagt vekt på at systemet skal være enkelt å bruke. Det er videre lagt vekt på å begrense plassforbruk og tidsforbruk.

Arne Venstad, Universitetet i Trondheim, Det Kgl. Norske Videnskabers Selskab, Museet, Erling Skakkesgt. 47B, N-7000 Trondheim.

INNHold

| | |
|--------------------------------------|----|
| Referat | 3 |
| 1. Beskrivelse av råmaterialet | 5 |
| 2. Brukerkrav til systemet | 6 |
| 3. Tekniske krav til systemet | 6 |
| 4. Beskrivelse av systemet | 7 |
| 4.1. Initieringsfasen | 8 |
| 4.2. Bruk av systemet | 9 |
| 5. Eksempler på bruk | 11 |
| Appendix A.1 - A.16, Programlister | |

1. BESKRIVELSE AV RÅMATERIALET

Somrene 1970, 1971, 1972 og 1973 ledet professor Svein Haftorn ved Zoologisk avdeling, DKNVS, Museet, en undersøkelse som hadde til hensikt å kartlegge forholdene i rugeperioden til forskjellige fugler. Ved hjelp av datalogger ble det foretatt registreringer i og omkring forskjellige fuglereder i området ved Klæbu. Dette ga som resultat store mengder råmateriale som seinere skulle behandles ved hjelp av EDB.

"Rådata" representerer registreringer av fysiske parametre som til eksempel temperatur, luftfuktighet, lysstyrke, nedbør osv. De fysiske parametrene kan deles i to grupper:

- 1) De faste målepunktene som registrerer klimaforhold. Disse er stort sett uforandret gjennom hele perioden.
- 2) De variable målepunktene som registrerer informasjon vedrørende én spesiell fugl (eggmålepunkter).

Dataloggerens registreringer ble stanset ut på papirhullbånd. Ved Universitetet i Oslo ble disse dataene overført til magnetbånd og samtidig transformert til et nytt format. Råmaterialet finnes nå på følgende bånd ved RUNIT's anlegg:

| Bånd nr. | Materiale | |
|----------|--------------|---------|
| 1471 | 1970 og 1971 | 2 filer |
| 1472 | 1972 | 1 fil |
| 1473 | 1973 | 1 fil |

Båndene må tilordnes med:

VASG,TIME INNTAPE.,C,XXXX

På disse båndene ligger materialet lagret som kortbilder og er skrevet i BCD-kode. Formatet avhenger av fra hvilket år materialet er innsamlet. I 1970, 1971 og 1972 ble det benyttet en datalogger med 60 målepunkter og 6 min. syklustid. I dette tilfellet lagres én syklus på fire kort i følgende format: <3(A1,16I5),A1,12I5> (ALGOL-notasjon). I 1973 ble det benyttet en logger med 26 målepunkter og syklustid 5 min. Her er da formatet: <A1,16I5,A1,10I5>.

2. BRUKERKRAV TIL SYSTEMET

En bruker vil ønske å trekke ut informasjon som angår en bestemt fugl fra det innloggede materialet. Informasjonen vil være f.eks. utskrift av temperaturregistreringene og klimaparametre, beregnet middelvei med standardavvik for bestemte tidsintervall eller grafisk plotting av bestemte parametre. Nødvendige parametre som må være kjent for å kunne trekke ut informasjon vedrørende en bestemt fugleregistrering, vil være tidsperioden for registreringen (år, dato og klokkeslett for start og slutt), hvilke målepunkter som ble benyttet for den spesielle fuglen, og hvilke målepunkter som ble benyttet til registrering av klimaforhold i den aktuelle perioden. De faste målepunktene vil bli gitt (default-) verdier av systemet dersom brukeren ikke angir andre verdier.

Det er ønskelig at brukeren bare skal behøve å kjenne til disse parametrene, som kan hentes fra notater gjort under den aktuelle registreringen. Det er systemets oppgave å holde rede på hvor de forskjellige data finnes og i hvilket format de er lagret. Bortsett fra at angivelse av målepunkt må være korrekt, skal brukeren heller ikke måtte ta hensyn til om det er ny eller gammel datalogger som er benyttet. Systemet må også kunne behandle eventuelle "huller" i datamaterialet og filtrere bort uakseptable verdier. Hvilke verdier som skal slippe gjennom filteret styres ved hjelp av parametre.

3. TEKNISKE KRAV TIL SYSTEMET

Det ville være svært uøkonomisk å anvende de informasjonssøkende operasjonene direkte på rådata i den form de foreligger. Systemet konverterer derfor råmaterialet over til et annet format som gjør de informasjonssøkende prosessene enklest mulig. Da det bare er behov for sekvensiell gjennom søking av materialet, er formatet som er valgt en sekvensiell, post-formatert fil. En post svarer til registreringene fra en full syklus på dataloggeren. Dette bør være den mest naturlige måten å organisere dataene på.

Datamaterialet fra en sommers registrering fyller ca. 300-600 trk på en Fastrandfil. Det er derfor viktig å ta hensyn til plassforbruk ved omformateringen av materialet. Henting av data fra sekundærlager innebærer

også tidkrevende inn/ut-operasjoner. Begge disse forhold gjør det nødvendig å lagebufret inn/ut-operasjoner i systemet.

Konverteringen av råmaterialet til postformaterte filer, som også innebærer en viss foredlingsprosess av data, er tenkt gjort én gang for alle i en initieringsfase av systemet. De foredlede datafilene lagres på magnetbånd og danner seinere datagrunnlaget for bruksfasen av systemet. Disse filene finnes på følgende bånd ved RUNIT's anlegg:

| Bånd nr. | Kopi | Registrering | Fil nr. | Lengde |
|----------|------|--------------|---------|----------|
| 1605 | 1606 | 1973 | 1 | 320 trk. |
| 1605 | 1606 | 1971 | 2 | 371 trk. |
| 1605 | 1606 | 1970 | 3 | 594 trk. |
| 1605 | 1606 | 1972 | 4 | 497 trk. |

4. BESKRIVELSE AV SYSTEMET

Mesteparten av programmene som utgjør systemet er skrevet i NU-ALGOL. Dette språket er valgt fordi det gir en grei og oversiktlig

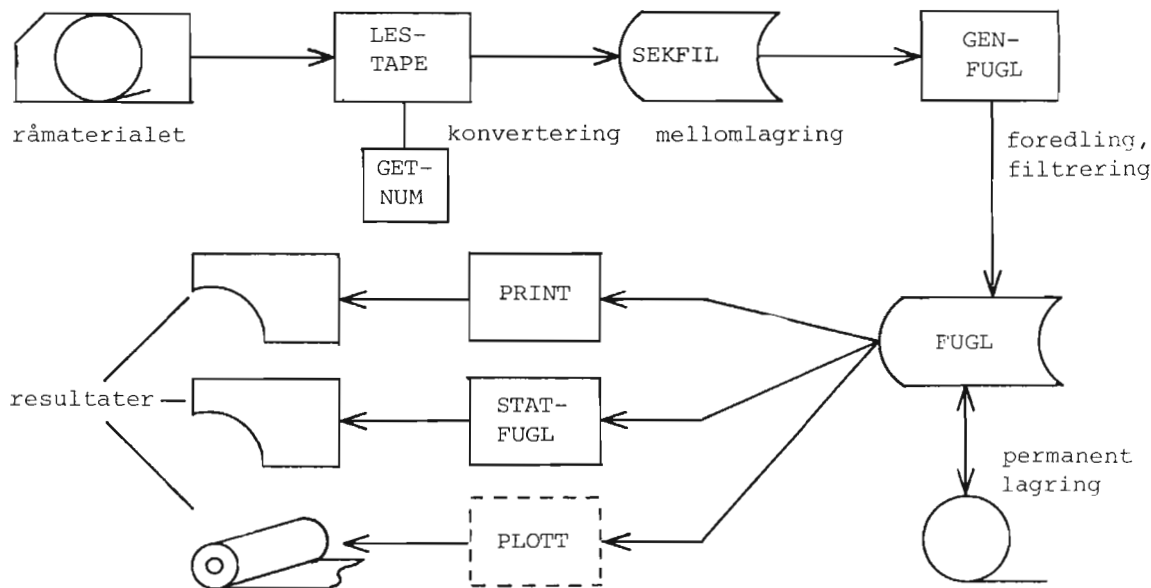


Fig. 1. ORNITOLOGG-systemet og dets komponenter.

notasjon, sikkerhet, og ved å bufre inn/ut-operasjonene på sekundærlager, gir det også tilstrekkelig effektiv kode. En subrutine (GETNUM) som leser fra båndene med råmaterialet måtte skrives i assembler. Dette var først og fremst på grunn av at båndene var skrevet på et annet dataanlegg med et annet format, men også for å få effektiv kode.

Fig. 1 viser systemets deler og den prosessen som dataene gjennomgår. Den øverste delen av figuren representerer initeringsfasen av systemet, mens den nederste representerer bruksfasen. Programmet PLOTT som skal tegne kurver på en grafisk terminal, er ennå ikke implementert.

4.1. Initieringsfasen

Denne delen overfører råmaterialet til foredlede data på sekvensielle ALGOL-filer, som lagres permanent på nye magnetbånd. Råmaterialet ligger lagret som kortbilder med blokk lengde på 100 kort. Grunnen til dette er at det tidligere ble benyttet et system av FORTRAN-programmer som brukte kortfiler som inndata. Da dette systemet var uoversiktlig, lite fleksibelt og svært dyrt å bruke, ble det besluttet å lage et nytt system.

Omformeringsprosessen skjer i to skritt med en mellomlagring på en Fastrandfil. Programmet LESTAPE foretar ved hjelp av subrutinen GETNUM en ren transformasjon av data fra kortbildeforamt til en sekvensiell Fastrandfil (SEKFIL) med poster som svarer til syklene på dataloggeren. Samtidig skriver programmet ut alle registrerte klokkeslett (ett for hver post) og postnumrene. På grunnlag av disse utskriftene og notater fra registreringsperioden kan en lokalisere kronologiske feil og "hull" i datamaterialet. Ved slike "hull" leveres gapet, dvs. antall datoskift, og nummeret på den tilsvarende posten som inn-data til programmet GENFUGL. Postlengden på SEKFIL er enten 60 eller 26, avhengig av om registreringen er fra før 1973 eller ikke.

Programmet GENFUGL leser den midlertidige filen SEKFIL, og ved hjelp av tilleggsopplysninger fra kort om startdato og eventuelle "hull" i materialet genereres en ny sekvensiell fil, FUGL. Poster med uakseptable klokkeverdier filtreres bort. Hvis nødvendig forandres klokketallet fra TTMM (ny logger) til TTMM. Som første attributt i hver post (ubrukt av logger) legges inn en verdi, DATO. Verdien framkommer som $100 * MND + DAG$, og utgangsverdien tas fra startdatoen for registreringene. Denne attributten

sammen med klokketallet som er siste attributt i posten, brukes som nøkler når en skal søke seg fram til et bestemt tidspunkt i datastrømmen. Eventuelle brudd i datastrømmen gir seg utslag i et tilsvarende sprang i verdien til DATO. Et normalt datoskifte (DATO økes med 1, 70, 71 eller 73) fås hver gang klokka passerer 2400. Normale intervaller for klokka er 5-6 min. Det tolereres imidlertid sprang på opptil MAXSPRANG, som er en parameter som leses inn av programmet. Er spranget større, forkastes alle de etterfølgende postene helt til det følger en post med klokketall innenfor det akseptable intervall. Det har vist seg at materialet inneholder en del dobbelte registreringer som enten har oppstått under logging eller ved senere behandling i Oslo (skummel FORTRAN-programmering!). Disse blir effektivt filtrert vekk som støy. Støyprosenten ligger på ca. 0.1-1%.

For å optimalisere plassforbruket, og for å spare tidkrevende inn/ut-operasjoner i NU-ALGOL, er det laget et sekundært I/U-system som bufrer lesing/skriving på sekundærfilene SEKFIL og FUGL. Systemet består av rutinene SKRIV, LES og CLOSE samt bufrene SKRIVBUF og LESBUF med tilhørende bufferpekere. Bufferlengden beregnes som den største mulige som er mindre enn 256 og som rommer et helt antall poster pluss statusord. 256 er den interne bufferlengde i NU-ALGOL.

Tabellen under viser innsparing i plassforbruk og tidsforbruk ved bufret I/U-system.

| | Uten bufring | | Med bufring | |
|------------------------|--------------|-------|-------------|-------|
| | | | | |
| Postlengde | 60 | 26 | 60 | 26 |
| Fyllingsprosent på fil | 23.4% | 10.2% | 96.5% | 94.1% |
| Inn/ut-kall | 100% | 100% | 25% | 11% |

Når programmet GENFUGL er kjørt, slettes filen SEKFIL. Filen FUGL kopieres ut på et nytt magnetbånd i rollout-format. Den danner så datagrunnlaget for de programmene som skal trekke ut informasjon fra materialet.

4.2. Bruk av systemet

Det er foreløpig skrevet to programmer, PRINT og STATFUGL, som begge på grunnlag av innleste parametre fra kort, søker ut og leser de

dataene på filen FUGL som vedrører en bestemt registrering av en fugl. Det første programmet lister ut i tabellform alle registreringer som er gjort for den spesielle fuglen sammen med klimaparametre, klokke og dato. STATFUGL beregner på grunnlag av de samme data middelverdier med tilhørende standardavvik, samt ekstremalverdier for hver time og for hvert døgn. (Timerverdier og døgnverdier beregnes uavhengig.) For begge programmer gjelder at de innleste parametrene må bestemme registreringsperioden nøyaktig ved klokkeslett, dag, måned for start og slutt, samt årstall. Videre må eggmålepunktene angis; hvor mange og hvilke numre. Dersom en ønsker å bytte ut noen av de faste målepunktene (klimaparametre) med f.eks. temperatur i reirbunn, luft i kassa, kjølespiral osv., angir en dette på de siste parameterkortene. Internt opererer programmene med en permutasjonsvektor, FINDEX, som har seks elementer. Disse er målepunktnumrene for de faste målepunktene. Initielt inneholder vektoren numrene for følgende faste målepunkt: utetemperatur, luftfuktighet, nedbør, lysstyrke, besøksteller, dummy. Verdiene av disse avhenger av om materialet er fra gammel eller ny logger.

Parameterkort 3 og 5 (se listing av program i Appendix) angir hvor mange og hvilke av de faste målepunktene som skal være med. På parameterkort nr. 7 (trenger ikke å være med) angis en indeks mellom 1 og 6 som er et av de faste målepunkt som skal byttes ut, nummeret til det nye målepunktet (et nummer på dataloggeren) og én 10-tegns streng som vil bli brukt som ny overskrift i tilsvarende tabellrubrikk. En kan skifte ut et vilkårlig antall av de faste målepunktene, og det trengs et parameterkort for hver utskifting.

Dersom ett eller flere målepunkt forandres i en registreringsperiode, må perioden deles opp, og programmene kjøres separat for hver delperiode.

Når det gjelder programmet STATFUGL, kan en i tillegg til vanlig utskrift få stanset ut alle timeverdiene på hullkort. I så fall må verdien på parameterkort nr. 6 være "TRUE".

Dersom den aktuelle registreringsperioden faller oppå et "hull" i datamaterialet, vil programmene gi beskjed om hvilke data som mangler. I programmet STATFUGL settes nedre og øvre grenseverdi for hvilke eggtemperaturer som aksepteres. Dette filtrerer vekk støy som ikke kommer med i middelverdiberegningene. Hvor stor støyprosenten er skrives ut til slutt.

Brukeren må sjøl sørge for å tilordne datafilen FUGL. For å spare dyr masselagerplass, er det funnet hensiktsmessig å ha bare materiale fra

ett år om gangen liggende på filen. Dette betyr at det blir brukerens ansvar å kopiere inn den riktige filen fra databåndet (se oversikt i kap. 3). Alle katalogiserte filer bør slettes så fort som mulig, da det er dyrt å ha slike liggende lenge.

5. EKSEMPLER PÅ BRUK

Alle de nevnte programmene ligger lagret i både symbolsk og absolutt form på bånd nr. 7643 (kopi 7755), fil nr. 2. I tillegg til datafilen FUGL må brukeren tilordne en fil og kopiere inn disse programmene på denne.

Vi skal se på et fullstendig eksempel. Vi ønsker å få ut full utskrift og middelverdiberegninger av de registreringene som ble gjort av svarthvit fluesnapper 1/70 i perioden kl. 1524 14. juni til kl. 1850 den 11. juli 1970. Det ble benyttet tre målepunkter til temperaturmålinger i egg, nr. 14, 24 og 44. Vi ønsker å få ut verdien på de fire første faste målepunktene, dvs. utetemperatur, luftfuktighet, nedbør og lysstyrke, og vi vil også ha resultatene fra STATFUGL stanset ut på hullkort. Følgende kortoppsett kan brukes:

```
1      "RUN,/TPC <BRUKERNR.>,<KJØRENR.>,<PRSJKT>,2,300/1000
2      "MSG,W *TEKST EVENTUELLE KORT*
3      "ASG,T PROG.
4      "ASG,T FUGL,,F//TRK/600
5      "ASG,T TAPE1,,C,7643
6      "MOVE TAPE1,,1
7      "COPI TAPE1,,PROG.
8      "ASG,T TAPE2,,C,1605
9      "MOVE TAPE2,,2
10     "COPY,S TAPE2,,FUGL.
11     "XQT PROG.PRINT
12     SVARTHVIT 1/70
13     1524 14 6 1850 11 7 1970
14     3
15     14 24 44
16     "XQT PROG.STATFUGL
17     SVARTHVIT 1/70
18     1524 14 6 1850 11 7 1970
19     3 4
20     14 24 44
21     1 2 3 4
22     TRUE
23     "FIN
```

| Kort nr. | Kommentar |
|----------|--|
| 1 | Det trengs rommelige grenser for tidsforbruk og sider ut! |
| 3-4 | Filene tilordnes som temporærfiler. Dersom det skal kjøres flere ganger lønner det seg å gjøre de permanente (bytt T med UP). Filen FUGL må være på 600 trk for å romme alle data. |
| 5-7 | Programmene kopieres inn fra fil nr. 2 på bånd 7643. |
| 8-10 | Datafilen kopieres inn fra fil nr. 3 på bånd 1605. |
| 11 | Første program utføres. |
| 12-15 | Datakort til PRINT. 1. kort: Identifikasjon i tabelloverskrift 2. kort: Tidsperiode for registrering 3. kort: Antall eggmålepunkt 4. kort: Eggmålepunkt, nummer på datalogger |
| 16 | Programmet STATFUGL utføres. |
| 17-22 | Datakort. 1. kort: Identifikasjon i tabelloverskrift 2. kort: Tidsperiode for registrering 3. kort: Antall eggmålepunkt, antall faste målepunkt 4. kort: Eggmålepunkt, nummer på datalogger 5. kort: Faste målepunkt, indekser i permutasjonsvektor 6. kort: Hullkort ut |

Det er å merke seg at programmet PRINT kan gi mange sider utskrift, slik at det kan lønne seg å kjøre de to programmene i to separate kjøringar. Dette tilsier også at en bør ha filene PROG og FUGL permanente over en kortere tid. Mest vil en spare ved å samle alle kjøringar en har behov for i en intens periode (gjerne med S-prioritet).

Dersom det kommer feilutskrift eller programmene terminerer unormalt, bør en sjekke parameterkortene nøye. Disse listes ut aller først av programmet.

APPENDIX A.1 - A.16

PROGRAMLISTER

ITHFAUN*PROG(1).LESTAPE

```

1     BEGIN
2     COMMENT
3     XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
4     XX
5     XX PROGRAMMET LESER 'RRDATA' FRA DATALOGGER SOM LIGGER PÅ BÅNDET 'INN-
6     XX TAPE' I BCD-KODE. SUBROUTINA GETNUM LESER ETT OG ETT TALL FRA BÅNDET.
7     XX DATAENE KONVERTERES TIL EN POSTFORMATTERT FIL ('SEKFIL') SOM SKRIVES
8     XX MED BUFRING FRA ALGOL. POSTFORMATET BESTEMMES AV HVILKET ÅR LOGGIN-
9     XX GEN ER FORETATT (GAMMEL LOGGER; POSTLENGDE=60, NY LOGGER; POSTLENGDE=
10    XX 26). DETTE ÅRSTALLET MÅ DERFOR GIS SOM INPUT TIL PROGRAMMET. FILENE
11    XX ('INNTAPE' OG 'SEKFIL') MÅ TILORDNES/OPPRETTES PÅ VANLIG MÅTE UTEN-
12    XX FOR PROGRAMMET.
13    XX
14    XX          *** INPUT: ***
15    XX
16    XX 1. KORT:  AAR          . ÅRSTALL FOR DATALOGGING (HELTALL)
17    XX
18    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
19    ;
20    INTEGER I,J,NSEK,AAR,PLENGTH,PMAX,BMAX,SKRIVINDEX;
21    EXTERNAL LIBRARY INTEGER PROCEDURE GETNUM(UT); LABEL UT;
22    READ(AAR);
23    PLENGTH:=IF AAR LSS 1973 THEN 60 ELSE 26;
24    PMAX:=ENTIER(249/PLENGTH);
25    BMAX:=PLENGTH*PMAX;
26    BEGIN
27        INTEGER ARRAY POST(1:PLENGTH),SKRIBUF(1:BMAX);
28
29        PROCEDURE SKRIV(FIL); VALUE FIL; STRING FIL;
30        BEGIN
31            INTEGER I,H;
32            COMMENT SKRIVINDEX PEKER PÅ FØRSTE LEDIGE POST I SKRIBUF;
33            IF SKRIVINDEX GEQ PMAX THEN
34                BEGIN
35                    COMMENT SKRIBUFFER ER FULL;
36                    WRITE(FILE(FIL),SKRIBUF);
37                    SKRIVINDEX:=0;
38                END;
39            H:=PLENGTH*SKRIVINDEX;
40            FOR I:=(1,1,PLENGTH) DO
41                SKRIBUF(H+I):=POST(I);
42            SKRIVINDEX:=SKRIVINDEX+1;
43        END *** SKRIV ***;
44
45        PROCEDURE CLOSE(FIL); VALUE FIL; STRING FIL;
46        BEGIN
47            INTEGER I,H;
48            H:=PLENGTH*SKRIVINDEX;
49            WRITE(FILE(FIL),FOR I:=(1,1,H) DO SKRIBUF(I));
50            WRITE(FILE(FIL),EOF);
51        END *** CLOSE ***;
52
53        WRITE(<<I5,X1>>,NSEK);
54        NYLINJE: FOR I:=(1,1,20) DO
55            BEGIN
56                NSEK:=NSEK+1;
57                FOR J:=(1,1,PLENGTH) DO POST(J):=GETNUM(SLUTT);
58                SKRIV('SEKFIL');
59                WRITE(<<I6>>,POST(PLENGTH));
60            END;
61        WRITE(<<A1,I5,X1>>,NSEK);

```

```
62      GOTO NYLINJE;
63      SLUTT: WRITE(<<A1.5,X10,I10,, SEKVENSER LEST, NORMAL TERMINERING.,A1
64              >>,NSEK);
65      CLOSE('SEKFIL');
66      END;
67      END;
```

THFAUN*PROG(1).GETNUM

```

1  . XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
2  . XX
3  . XX SUBROUTINA GETNUM KALLES FRA ET ALGOL HOVEDPROGRAM HVOR DEN ER
4  . XX DEKLARERT SOM
5  . XX
6  . XX     EXTERNAL LIBRARY INTEGER PROCEDURE GETNUM(UT); LABEL UT;
7  . XX
8  . XX RUTINA LESER ETT OG ETT HELTALL FRA BÅNDET 'INNTAPE' MED FORMAT
9  . XX I5. (ET FELT MED 5 BLANKE TEGN IGNORERES.) DETTE BÅNDET ER SKREVET
10 . XX I BCD-KODE, OG TALLENE LIGGER LAGRET I KORTBILDER A 80(1) TEGN.
11 . XX VED FORMATFEIL LEVERER GETNUM TALLET -9999, OG NÅR EOF
12 . XX PÅTREFTES RETURNERER KONTROLLEN TIL PARAMETER-ETIKETTEN I HOVED-
13 . XX PROGRAMMET.
14 . XX
15 . XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
16 I(1)  AXRI
17 GETNUM* S      AO,RETUR      . RETURADRESSE VED EOF
18           SZ      TALL
19           SZ      A5          . HJELPEREGISTER
20           L,U     A4,1        . FORTEGN
21           LR,U    R1,4        . GJØR KLAR TIL Å LESE 5 TEGN
22 NYSIF      SLJ    GETCH      . NYTT TEGN I AO
23           TNE,U   AO,5        . BLANK?
24           J       UT          . JA, IGNORER
25           TNE,U   AO,'+'      . + ?
26           J       UT          . JA, IGNORER
27           TE,U    AO,'-'      . - ?
28           J       I+3        . NEI, FORTSETT
29           MSI,XU  A4,-1      . JA SETT FORTEGN
30           J       UT          . OG LES NESTE TEGN
31           TLE,U   AO,'0'      .
32           J       FEIL        . IKKE SIFFER
33           TG,U    AO,072      .
34           J       FEIL        . IKKE SIFFER
35           L,U     A5,1        . SIFFER I FELTET
36           AN,U    AO,'0'      . SIFFER I AO
37           L       A1,TALL
38           MSI,U   A1,10
39           A       AO,A1
40           S       AO,TALL
41 UT         JGD    R1,NYSIF    . AKKUMULERER TALLET
42           JZ      A5,GETNUM+1 . HENT 5 SIFFER
43           L       AO,TALL      . IKKE SIFFER I FELTET GR TIL NESTE
44           MSI     AO,A4        . RETURNER TALLET I AO
45           J       O,X11        . MED RIKTIG FORTEGN
46           DUMLES SLJ    GETCH  . RETURNER TIL HP
47 FEIL       JGD    R1,DUMLES   . DUMMY
48           L,XU    AO,-9999     . MÅ LESE RESTEN AV DE 5 TEGNENE
49           J       O,X11        . LEVER DETTE TALLET VED FEIL
50 GETCH      I-I          . HENTER ET TEGN FRA WORD
51           L       A1,WORD
52           TNZ     A1
53           SLJ    NWRD          . ER ORDET TØMT?
54           OK     SZ      AO     . JA, HENT NYTT ORD
55           LDSL   AO,6        . KLAR TIL Å HENTE TEGN
56           S       A1,WORD
57           J       *GETCH
58 NWRD       I-I          .
59           L       AO,WCNT      . PEKER TIL SIST LESTE ORD I BUF
60           TG      AO,MAX
61           SLJ    LES          . ER BUF TØMT?
                       . JA, FYLL DEN IGJEN

```

```

62      A,U   AO,1           . INKREMENTER PEKER
63      S     AO,WCNT       . PEKER FORTSATT I AO
64      L     A1,BUF,AD     . HENT NYTT ORD
65      J     *NWRD         . RETUR-HOPP
66      LES   I-1
67      L,U   AO,PK
68      ER    IOWI
69      L,H2  A3,PK+3       . ANTALL ORD OVERFØRT
70      TNE,U A3,1
71      J     GOON+2
72      AN,U  A3,1           . JUSTERER
73      S     A3,MAX        . PEKER PÅ SISTE SISTE ORD I BUF
74      . LISNAP 'DUMP',0,1400,BUF
75      L,S1  AO,PK+3       . TEST PÅ STATUS
76      TNE,U AO,1         . EOF?
77      J     *RETUR        . JA, RETURNER TIL EOF-LABEL I HP
78      TE,U  AO,4         . ILL. FRAME COUNT?
79      J     GOON          . NEI, TEST VIDERE
80      L,U   AO,5
81      AN,S3 AO,PK+3       . JA, FIKS SISTE ORD I BUF
82      L     A2,BUF,A3     . ANTALL UBRUKTE BYTES I SISTE ORD
83      LR    R2,AD         . SISTE ORD I BUF
84      SSC   A2,6
85      JGD   R2,1-1        . SKIFT DE UBRUKTE BYTES
86      LR    R2,AD         . UT TIL HØGRE
87      LSSL  A2,6
88      JGD   R2,1-1        . SKIFT TILBAKE MED
89      S     A2,BUF,A3     . NULL-UTFYLLING
90      J     I+3           . LEGG ORDET TILBAKE
91      GOON  TZ            . 0 ?
92      ER    ERR1         . NEI, ANNEN FEIL
93      L,S1  AO,BUF       . UNDERSØK 1. ORD I BLOKKA
94      TE,U  AO,076       . 1. BYTE=76?
95      J     OKUT         . NEI, START PÅ 1. ORD
96      . LISNAP 'DUMP1',02,14,BUF
97      J     LES+1
98      OKUT  L,XU  AO,-1
99      J     *LES
100     SKIPB* LR,U   R1,310
101      L,U   AO,PK
102      ER    IOWI
103      JGD   R1,1-2
104      J     O,X11
105     I(O)
106     RETUR  +0
107     WORD   +0
108     WCNT   +0
109     MAX    +0
110     TALL   +0
111     FBW    +0416460606005
112     PK     'INNTAP'
113     'E     '
114     *0
115     *02000,0
116     *1400,BUF
117     BUF    RES 1400
118     END

```

```

ITHFAUN*PROG(1).GENFUGL
1  BEGIN
2  COMMENT
3  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
4  %%
5  %% PROGRAMMET LESER DATASEKVENSER FRA FILEN 'SEKFI'. KLOKKETALLET
6  %% KONVERTERES TIL TTMM OG EVENTUELLE SEKVENSER MED 'GALE' KLOKKETALL
7  %% FILTRERES BORT. STARTDATOEN FOR MÅLINGENE LESES INN OG PROGRAMMET
8  %% LEGGER INN ATTRIBUTTEN 100*MND+DAG SOM POST(1) SAMTIDIG SOM DATOEN
9  %% INKREMENTERES. DERSOM MATERIALET INNEHOLDER NATURLIGE DISKONTINU-
10 %% ERLIGHETER, MÅ DISSE GIS VED INPUT-PARAMETRE FOR AT DE ETTERFØLGENDE
11 %% SEKVENSENE SKAL PASSERE KLOKKEFILTERET. DE FILTRERTE OG
12 %% KRONOLOGISK RIKTIGE SEKVENSENE LEGGES UT PÅ FILEN 'FUGL' MED BUFRET
13 %% SKRIVING.
14 %% POST(DUMMY) SETTES LIK NULL OG BRUKES KUN AV UTSKRIFTSPROGRAM-
15 %% MENE AV TEKNISKE GRUNNER.
16 %%
17 %%          *** INPUT: ***
18 %%
19 %% 1. KORT:  DAG,MND,AAR          . STARTDATO FOR MÅLESERIE
20 %%
21 %% 2. KORT:  ANTBRUDD            . ANTALL DISKONTINUERLIGHETER.
22 %%
23 %% 3. KORT:  BRUDDSEK(1;ANTBRUDD). SEKV. NR. TIL BRUDD
24 %%
25 %% 4. KORT:  BRUDDOGN(1;ANTBRUDD). DATO-SPRANG VED BRUDD
26 %%
27 %% 5. KORT:  MAXSPRANG           . STØRSTE TIDSPRANG SOM TOLERERES
28 %%
29 XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
30 ;
31 INTEGER SKRIVINDEX,LESINDEX,PMAX,BMAX,PLENGTH,DUMMY;
32 INTEGER I,DATO,DAG,MND,AAR,NSEK,FSEK,T,KLOKKE,MAXSPRANG,ANTBRUDD;
33
34 READ(DAG,MND,AAR);
35 DUMMY:=IF AAR LSS 1973 THEN 2 ELSE 12;
36 PLENGTH:=IF AAR LSS 1973 THEN 60 ELSE 26;
37 READ(ANTBRUDD); COMMENT ANTALL BRUDD I DATAREGISTRERINGER;
38 PMAX:=ENTIER(249/PLENGTH);
39 BMAX:=PLENGTH*PMAX;
40 BEGIN
41 INTEGER ARRAY MDAG(1;12),POST(1;PLENGTH),SKRIVBUF,LESBUF(1;BMAX);
42 INTEGER ARRAY BRUDDSEK,BRUDDOGN(1;ANTBRUDD);
43
44 PROCEDURE SKRIV(FIL); VALUE FIL; STRING FIL;
45 BEGIN
46 INTEGER I,H;
47 COMMENT SKRIVINDEX PEKER PÅ 1. LEDIGE POST I SKRIVBUF
48 SOM SKAL SKRIVES;
49 IF SKRIVINDEX GEQ PMAX THEN
50 BEGIN
51 COMMENT SKRIVBUF ER FULL;
52 WRITE(FILE(FIL),SKRIVBUF);
53 SKRIVINDEX:=0;
54 END;
55 H:=PLENGTH*SKRIVINDEX;
56 FOR I:=(1,1,PLENGTH) DO
57 SKRIVBUF(I+H):=POST(I);
58 SKRIVINDEX:=SKRIVINDEX+1;
59 END *** SKRIV ***;
60
61 PROCEDURE LES(FIL); VALUE FIL; STRING FIL;

```

```

62 BEGIN
63   INTEGER I,H;
64   COMMENT LESINDEX PEKER PAA 1. LEDIGE POST I LESBUF
65     SOM SKAL LESES;
66   IF LESINDEX GEQ PMAX THEN
67     BEGIN
68       COMMENT LESBUF ER TØMT;
69       READ(FILE(FIL),LESBUF,SLUTT,SLUTT,FEIL);
70       LESINDEX:=0;
71     END;
72   H:=PLENGTH+LESINDEX;
73   FOR I:=(1,1,PLENGTH) DO
74     POST(I):=LESBUF(H+I);
75   LESINDEX:=LESINDEX+1;
76 END *** LES ***;
77
78 PROCEDURE CLOSE(FIL); VALUE FIL; STRING FIL;
79 BEGIN
80   INTEGER I,H;
81   H:=PLENGTH+SKRIVINDEX;
82   WRITE(FILE(FIL),FOR I:=(1,1,H) DO SKRIVBUF(I));
83   WRITE(FILE(FIL),EOF);
84 END *** CLOSE ***;
85
86   PROCEDURE TTMM;
87   BEGIN
88     INTEGER K1,K2;
89     K1:=ENTIÉR(POST(PLENGH)/1000);
90     K2:=POST(PLENGH)-1000*K1;
91     POST(PLENGH):=100*K1+K2;
92   END *** TTMM ***;
93
94   PROCEDURE INCDATO(N); VALUE N; INTEGER N;
95   BEGIN
96     DAG:=DAG+N;
97     IF DAG GTR MDAG(MND) THEN
98       BEGIN
99         DAG:=DAG-MDAG(MND);
100        MND:=MND+1;
101      END;
102   END *** INCDATO ***;
103
104   LESINDEX:=PMAX;
105   READ(BRUDDSEK); COMMENT SEKVENSNR. TIL BRUDD;
106   READ(BRUDDØGN); COMMENT ANTALL BRUDDØGN;
107   READ(MAXSPRANG); COMMENT MAKS. TILLATT TIDSPRANG I MÅLINGENE;
108   FOR I:=1,3,5,7,8,10,12 DO MDAG(I):=31;
109   FOR I:=4,6,9,11 DO MDAG(I):=30;
110   MDAG(2):=IF MOD(AAR,4) EQL 0 THEN 29 ELSE 28;
111
112   FIRST: NSEK:=NSEK+1;
113   LES('SEKFIL');
114   IF AAR GEQ 1973 THEN TTMM;
115   T:=POST(PLENGH);
116   IF T GEQ 0 AND T LSS 2400 THEN
117     BEGIN
118       COMMENT STARTPOST OK.;
119       DATO:=100*MND+DAG;
120       WRITE(<<'NY PERIODE STARTER ',I2,'/'',I1,' KL.',D7.2,' MED SEKVENSNR
121 .',I6,A3.3>>,DAG,MND,T/100,NSEK);
122       KLOKKE:=T;
123       POST(1):=DATO;

```



```

124         POST(DUMMY):=0;      COMMENT DUMMY=FELT;
125         SKRIV('FUGL');
126     END ELSE
127     BEGIN
128         FSEK:=FSEK+1;
129         WRITE('FEIL KLOKKE I STARTPOST:',POST(PLENGTH));
130         GOTO FIRST;
131     END;
132 NYSEK:  FOR I:=(1,1,ANTBRUDD) DO
133     IF NSEK+1 EQL BRUDDSEK(I) THEN
134     BEGIN
135         INCDATO(BRUDDOGN(I));
136         GOTO FIRST;
137         COMMENT FORTSETTER MED NY KLOKKE OG DATO;
138     END;
139     NSEK:=NSEK+1;
140     LES('SEKFIL');
141     IF AAR GEQ 1973 THEN TTMM;
142     T:=POST(PLENGTH);
143     IF T GEQ 0 AND T LSS 2400 THEN
144     BEGIN
145         IF T LSS KLOKKE THEN T:=T+2400;
146         IF T-KLOKKE LEQ MAXSPRANG THEN
147         BEGIN
148             COMMENT TIDSPUNKTET GODTAS MED MAKS. SPRANG MAXSPRANG;
149             KLOKKE:=POST(PLENGTH);
150             IF T NEQ KLOKKE THEN
151             BEGIN
152                 COMMENT DØGSKIFTE;
153                 INCDATO(1);
154                 DATO:=100*MND+DAG;
155             END;
156             POST(1):=DATO;
157             POST(DUMMY):=0;      COMMENT DUMMY=FELT;
158             SKRIV('FUGL');
159             GOTO NYSEK;
160         END;
161     END;
162     COMMENT KRONOLOGISK FEIL MED POST;
163     WRITE(('<<'FEIL KLOKKE I SEKVENS NR.',I6,D10.2,A1.1>>',
164         NSEK,POST(PLENGTH)/100);
165     FSEK:=FSEK+1;
166     GOTO NYSEK;
167 FEIL:  WRITE('GRUMS PÅ FILEN SEKFIL I SEKV. NR.',NSEK);
168 SLUTT: WRITE(('<<I6,' SEKVENSER LEST,',I6,' SEKVENSER FORKASTET.',
169     D6.2,' % STØY',A1.1>>',NSEK,FSEK,100*FSEK/NSEK);
170     CLOSE('FUGL');
171     END;
172 END;

```

```

THFAUN*PROG(1).PRINT
1  BEGIN
2  COMMENT
3  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
4  %%
5  %% PROGRAMMET SKRIVER UT DE REGISTRERTE MÅLINGER FOR EN SPESIFISERT
6  %% FUGL. FUGLEN SPESIFISERES VED INPUT-PARAMETRE.
7  %%
8  %%          *** INPUT; ***
9  %%
10 %% 1. KORT:  SPECIES          . STRENG PÅ 20 TEGN SOM
11 %%                               . INNEHØLDER FUGLENAVN.
12 %%
13 %% 2. KORT:  KL,DAG,MND,SKL,SDAG,SMND,AAR. PERIODEGRENSE SAMT ÅR.
14 %%
15 %% 3. KORT:  NE                . ANTALL EGGMÅLEPKT.
16 %%
17 %% 4. KORT:  EMPKT(1;NE)      . NE MÅLEPUNKT PUNCHES
18 %%                               . I FRITT FORMAT.
19 %%                               . (MÅLEPKT.=NR. PÅ DATALØGGER)
20 %%
21 %% 5. KORT:  I,RPKT,FELT      . ALTERNATIVT MÅLEPUNKT
22 %%                               . DEFINERES MED EN INDEX,
23 %%                               . ET LØGGER-PUNKT OG EN TI
24 %%                               . TEGNS STRENG. (TO HELTALL OG
25 %%                               . EN STRENG). KORTET KAN
26 %%                               . GJENTAS VILKÅRLIG MANGE
27 %%                               . GÅNGER, AVHENGIG AV HVOR
28 %%                               . MANGE AV DE FASTE MÅLEPUN-
29 %%                               . KTENE EN ØNSKER Å BYTTE UT.
30 %%                               . SE KOMMENTAR I STATFUGL.
31 XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
32 ;
33   INTEGER I, KL, DAG, MND, SKL, SDAG, SMND, AAR, DATO, SDATO, NE, RPKT, PMAX, BMAX,
34   LESINDEX, PLENGTH, TP, DUMMY, DOGN, N, DIFF, J;
35   STRING HEADING(132), FELT(10);
36   INTEGER ARRAY FINDEX, EMPKT(1;6);
37   INTEGER ARRAY MDAG(1;12);
38   STRING SPECIES(20);
39   STRING DM(4);
40   FORMAT F(D10.2,X3,S2,' ','S1,'-',I2,I6,D10.2,3V10,2U10.2,X3,:NE:(D7.2),
41   A1);
42   READ(<<A,S20>>,SPECIES);
43   READ(KL,DAG,MND,SKL,SDAG,SMND,AAR);
44   READ(NE);
45   READ(FOR I:=(1,1,NE) DO EMPKT(I));
46   WRITE(<<X45,'*** INPUT; ***',A2,3>>);
47   WRITE(<<'1. KORT: ',X5,S20,A1>>,SPECIES);
48   WRITE(<<'2. KORT: ',X5,I6,A1>>,KL,DAG,MND,SKL,SDAG,SMND,AAR);
49   WRITE(<<'3. KORT: ',X5,I6,A1>>,NE);
50   WRITE(<<'4. KORT: ',X5,:NE:(V6),A1>>,FOR I:=(1,1,NE) DO EMPKT(I));
51   IF AAR LSS 1973 THEN
52     BEGIN
53       COMMENT GAMMEL DATALØGGER;
54       FINDEX(1):=25;    COMMENT UTETEMPERATUR;
55       FINDEX(2):=48;    COMMENT LUFTFUKTIGHET;
56       FINDEX(3):=35;    COMMENT NEDBØR;
57       FINDEX(4):=58;    COMMENT LYSSTYRKE;
58       FINDEX(5):=53;    COMMENT BESØKSTELLER;
59       PLENGTH:=60;
60       TP:=100;
61       DUMMY:=2;

```

```

62     END ELSE
63     BEGIN
64         COMMENT NY DATALOGER;
65         FINDEX(1):=4;    COMMENT UTETEMPERATUR;
66         FINDEX(2):=13;   COMMENT LUFTFUKTIGHET;
67         FINDEX(3):=11;   COMMENT NEDBØR;
68         FINDEX(4):=14;   COMMENT LYSSTYRKE;
69         FINDEX(5):=12;   COMMENT IKKE I BRUK PÅ 73-LOGGER;
70         PLENGTH:=26;
71         TP:=10;
72         DUMMY:=12;
73         FOR I:=(1,1,NE) DO EMPKT(I):=EMPKT(I)+1;
74         COMMENT NY LOGGER NUMMERERES FRA 0!;
75         END;
76         FINDEX(6):=DUMMY;
77         WRITE(CORE(HEADING),<<'          KL.          DATO DØGN          UTET.',
78             L.FUKT.  NEDBØR  LYSST.  BES.FR.  REIRT.  ',NE:('EGG(',12,
79             ')'),A1>>,FOR I:=(1,1,NE) DO EMPKT(I));
80     ALTPKT; READ(I,RPKT,FELT,FORTS);
81         FINDEX(I):=RPKT;
82         HEADING(10*(I+1)+7,10):=FELT;
83         WRITE(<<I1,', KORT:',X5,2I6,S10,A1>>,J+5,I,RPKT,FELT);
84         J:=J+1;
85         GOTO ALTPKT;
86     FORTS; PMAX:=ENTIER(249/PLENGTH);
87         BMAX:=PLENGTH*PMAX;
88         BEGIN
89             INTEGER ARRAY POST(1:PLENGTH),LESBUF(1:BMAX);
90
91
92         PROCEDURE LES(FIL,UT,FE); VALUE FIL; STRING FIL;
93             LABEL UT,FE;
94         BEGIN
95             INTEGER I,H;
96             COMMENT LESINDEX PEKER PÅ FØRSTE POST I LESBUF SOM
97                 SIKAL LESES;
98             IF LESINDEX GEQ PMAX THEN
99                 BEGIN
100                     COMMENT LESBUF ER TØMT;
101                     READ(FILE(FIL),LESBUF,UT,UT,FE);
102                     LESINDEX:=0;
103                 END;
104             H:=PLENGTH+LESINDEX;
105             FOR I:=(1,1,PLENGTH) DO
106                 POST(I):=LESBUF(I+H);
107             LESINDEX:=LESINDEX+1;
108         END *** LES ***;
109
110     PROCEDURE POSISJONER(FIL,N1,N2,UT,FE); VALUE FIL,N1,N2;
111         STRING FIL; INTEGER N1,N2; LABEL UT,FE;
112     BEGIN
113         INTEGER H;
114         H:=BMAX-PLENGTH+1;
115         LESINDEX:=0;
116     OM; READ(FILE(FIL),LESBUF,UT,UT,FE);
117         IF LESBUF(H) LSS N1 THEN GOTO OM;
118         IF LESBUF(BMAX) LSS N2 THEN GOTO OM;
119         FOR LESINDEX:=LESINDEX+1 WHILE LESINDEX LSS PMAX AND
120             LESBUF((LESINDEX+1)*PLENGTH) LSS N2 DO;
121     END *** POSISJONER ***;
122
123     FOR I:=1,3,5,7,8,10,12 DO MDAG(I):=31;

```

```

124     FOR I:=4,6,9,11 DO MDAG(I):=30;
125     MDAG(2):=28;
126     DATO:=100*MND+DAG;
127     SDATO:=100*SMND+SDAG;
128     DOGN:=1;
129     N:=6;
130     WRITE(<<E1,X30,'*** 'S20,' FRA 'I2,'/'I1,' TIL 'I2,'/'I1,I6,' *
131 ***',A1,5>>,SPECIES,DAG,MND,SDAG,SMND,AAR);
132     WRITE(<<S132,A1.1>>,HEADING);
133     POSISJONER('FUGL',DATO,KL,NOTFOUND,FEIL);
134     NOK: IF DATO GTR LESBUF(1) THEN LES('FUGL',SLUTT,FEIL);
135     SKRIV: LES('FUGL',SLUTT,FEIL);
136     N:=N+1;
137     IF DATO NEQ POST(1) THEN
138     BEGIN
139     DIFF:=MOD(POST(1),100)-MOD(DATO,100);
140     IF DIFF LSS 0 THEN DIFF:=DIFF+MDAG(MND);
141     IF DIFF NEQ 1 THEN
142     BEGIN
143     I:=ENTIER(POST(1)/100);
144     J:=MOD(POST(1),100);
145     WRITE(<<'*** DATA MANGLER FOR PERIODEN 'I2,'/'I1,' - '
146     I2,'/'I1,' ***',A2,2>>,MOD(DATO,100),ENTIER(DATO/100),
147     J,I);
148     END;
149     DOGN:=DOGN+DIFF;
150     DATO:=POST(1);
151     END;
152     DM:=POST(1);
153     WRITE(F,POST(PLENGTH)/100,DM(2,2),DM(1,1),AAR-1900,DOGN,POST(FINDEX
154 (1))/TP,FOR I:=(2,1,6) DO POST(FINDEX(I))/100,FOR I:=(1,1,NE) DO
155     POST(EMPKT(I))/TP);
156     IF MOD(N,60) EQL 0 THEN WRITE(<<E1,S132,A1.1>>,HEADING);
157     IF POST(1) LSS SDATO THEN GOTO SKRIV;
158     IF POST(PLENGTH) LSS SKL THEN GOTO SKRIV;
159     GOTO SLUTT;
160     NOTFOUND: WRITE('DATA IKKE FUNNET'); GOTO SLUTT;
161     FEIL: WRITE('FEIL I FIL-I/O');
162     SLUTT:
163     END;
164     END;

```

THFAUN*PROG(1).STATFUGL

```

1 BEGIN
2 COMMENT
3
4
5 %% PROGRAMMET PLUKKER UT EN MÅLESERIE FOR EN SPESIFISERT FUGL FRA FILEN
6 %% 'FUGL'. PÅ GRUNNLAG AV DISSE DATA BEREGNES MIDDELVERDIER, STANDARD-
7 %% AVVIK OG EKSTREMALVERDIER FOR HVER TIME OG HVERT DØGN I PERIODEN.
8 %% INPUT-PARAMETRE MÅ SPESIFISERE PERIODEN (STARTTID OG SLUTTID), MÅLE-
9 %% PUNKT(ER) FOR EGG, OG HVOR MANGE AV DE FASTE MÅLEPUNKTENE SOM SKAL
10 %% VÆRE MED. DE FASTE MÅLEPUNKTENE ER UTETEMPERATUR(1), LUFTFUKTIGHET
11 %% (2), NEDBØR(3), LYSSTYRKE(4), BESØKSFREKVEN(5) OG REIRBUNNTEMPERA-
12 %% TUR(6). HVILKE AV DISSE SOM SKAL VÆRE MED ANGIS PÅ ET DATAKORT
13 %% MED TALLENE I PARENTESENE.
14 %% DE FASTE MÅLEPUNKTENE KAN BYTTES UT SÅ LANGT DET ER ØNSKELIG.
15 %% PARAMETERKORT NR. 7 ANGIR TO INDEKSER OG EN STRENG. DEN FØRSTE
16 %% INDEKSEN ANGIR HVILKET MÅLEPUNKT SOM SKAL BYTTES UT, DEN ANDRE
17 %% ANGIR MÅLEPUNKTNUMMERET PÅ DATALOGEREN. EN 10 TEGNS STRENG
18 %% ANGIR RIKTIG OVERSKRIFT I UTSKRIFTSTABELLEN (PUNCHES MED APOSTRØFER).
19 %% PARAMETERKORTET GJENTAS SÅ MANGE GANGER SOM ANTALL MÅLEPUNKT EN
20 %% ØNSKER Å BYTTE UT. HVIS DET IKKE FINNES NOE PARAMETERKORT AV
21 %% DENNE TYPEN, BRUKES DE FASTE MÅLEPUNKTENE SOM ANGITT.
22 %%
23 %% PROGRAMMET BEREGNER BARE MIDDELVERDIENE AV DISSE STØRRELSENE. FOR
24 %% EGGTEMPERATURENE BEREGNES MIDDELVERDI, STANDARDAVVIK, MINIMAL- OG
25 %% MAKSIMALVERDI, SAMT ANTALL MÅLINGER I HVERT BEREGNINGSGRUNNLAG.
26 %% PÅ DATAKORT NR. 6 ANGIS DET OM EN ØNSKER Å FÅ PUNCHET TIME-
27 %% VERDIENE UT PÅ KORT.
28 %%
29 %% *** INPUT ***
30 %%
31 %% 1. KORT: SPECIES . STRENG PÅ 20 TEGN SOM
32 %% . INNEHOLDER FUGLENAVN.
33 %%
34 %% 2. KORT: KL,DAG,MND,SKL,SDAG,SMND,AAR . PERIODEGRENSE SAMT ÅR.
35 %% 3. KORT: NE,NF . ANTALL EGGMÅLEPKT., ANTALL
36 %% . FASTE MÅLEPUNKT.
37 %%
38 %% 4. KORT: EMPKT(1:NE) . NE MÅLEPUNKT
39 %% . PUNCHES I FRITT FORMAT
40 %% . (MÅLEPKT=NR. PÅ DATALOGER.)
41 %%
42 %% 5. KORT: FMPKT(1:NF) . FASTE MÅLEPUNKTER. (INDEKS 1
43 %% . TIL 6).
44 %%
45 %% 6. KORT: KORTUT . TRUE GIR KORT UT.
46 %% . FALSE GIR
47 %% . INGEN KORT UT
48 %%
49 %% 7. KORT: I,RPKT,FELT . TO HELYTALL OG EN STRENG PÅ
50 %% . 10 TEGN ANGIR ET ALTERNATIVT
51 %% . MÅLEPUNKT. KORT 7 KAN GJEN-
52 %% . TAS VILKÅRLIG MANGE GANGER
53 %%
54
55 ;
56 BOOLEAN KORTUT;
57 INTEGER I,K,NE,NF,TANT,DANT,TIME,DOGN,DATO,SDATO,REFDATO,DAG,SDAG,
58 MND,SMND,KL,SKL,AAR,RPKT,PLENGTH,LESINDEX,PMAX,BMAX,TP,PT;
59 INTEGER OKN,FEILN,DUMMY,DIFF;
60 INTEGER ARRAY EMPKT,FMPKT,FINDEX,NT,ND(1:6);
61 INTEGER ARRAY MDAG(1:12);

```

```

62 REAL ARRAY TA,DA,TM,DM(1:6);
63 REAL TAKK,DAKK,P,TKVSUM,DKVSUM,TMAX,TMIN,DMAX,DMIN,TMID,TVAR,DMID,
64     DVAR;
65 STRING SPECIES(20),FELT(10);
66 STRING SH1(1:2),SH2(132),SH3(132);
67 FORMAT F1(17,'-',12,15,'/',1,1,'-',12,U10.2,3V10,2U10.2,X2,4D10.2,I10,
68     A1);
69 FORMAT F2(X10,15,'/',1,1,'-',12,U10.2,3V10,2U10.2,X2,4D10.2,I10,A1);
70 FORMAT FPUNCH(4I5,D7.2,2I5,5D7.2,I5,A1);
71 READ(<<A,S20>>,SPECIES);
72 READ(KL,DAG,MND,SKL,SDAG,SMND,AAR);
73 READ(NE,NF);
74 READ(FOR I:=(1,1,NE) DO EMPKT(I));
75 READ(FOR I:=(1,1,NF) DO FMPKT(I));
76 READ(KORTUT);
77     DUMMY:=IF AAR LSS 1973 THEN 2 ELSE 12;
78     WRITE(<<X45,'*** INPUT: ***',A2.3>>);
79     WRITE(<<'1. KORT:',X5,S20,A1>>,SPECIES);
80     WRITE(<<'2. KORT:',X5,7I6,A1>>,KL,DAG,MND,SKL,SDAG,SMND,AAR);
81     WRITE(<<'3. KORT:',X5,2I6,A1>>,NE,NF);
82     WRITE(<<'4. KORT:',X5,NE:(V6),A1>>,FOR I:=(1,1,NE) DO EMPKT(I));
83     WRITE(<<'5. KORT:',X5,NF:(V6),A1>>,FOR I:=(1,1,NF) DO FMPKT(I));
84     WRITE(<<'6. KORT:',X5,B5,A1>>,KORTUT);
85 IF AAR LSS 1973 THEN
86 BEGIN
87     COMMENT GAMMEL DATALOGGER;
88     FINDEX(1):=25;     COMMENT UTETEMPERATUR;
89     FINDEX(2):=48;     COMMENT LUFTFUKTIGHET;
90     FINDEX(3):=35;     COMMENT NEDBØR;
91     FINDEX(4):=58;     COMMENT LYSSTYRKE;
92     FINDEX(5):=53;     COMMENT BESØKSTELLER;
93     PLENGTH:=60;
94     TP:=100;
95     PT:=1;
96 END ELSE
97 BEGIN
98     COMMENT NY DATALOGGER;
99     FINDEX(1):=4;     COMMENT UTETEMPERATUR;
100    FINDEX(2):=13;     COMMENT LUFTFUKTIGHET;
101    FINDEX(3):=11;     COMMENT NEDBØR;
102    FINDEX(4):=14;     COMMENT LYSSTYRKE;
103    FINDEX(5):=12;     COMMENT BRUKES IKKE PÅ 73-LOGGER;
104    PLENGTH:=26;
105    TP:=10;
106    PT:=10;
107    FOR I:=(1,1,NE) DO EMPKT(I):=EMPKT(I)+1;
108    END;
109    FINDEX(6):=DUMMY;
110    SH2:='          TIME          DATO          UTET.  L.FUKT.  NEDB.  LYSST.  B
111    ES.FR.  REIRT.          EGGTEMPERATUR';
112    ALTPKT; READ(I,RPKT,FELT,START);
113    FINDEX(I):=RPKT;
114    SH2(10*(I+1)+1,10):=FELT;
115    WRITE(<<'1.',KORT:',X5,2I6,S10,A1>>,K+7,I,RPKT,FELT);
116    K:=K+1;
117    GOTO ALTPKT;
118    START:=PMAK:=ENTIER(249/PLENGTH);
119    BMAK:=PMAK+PLENGTH;
120    FOR I:=1,3,5,7,8,10,12 DO MDAG(I):=31;
121    FOR I:=4,6,9,11 DO MDAG(I):=30;
122    MDAG(2):=28;
123    BEGIN

```

```

124     INTEGER ARRAY POST(1:PLENGTH),LESBUF(1:BMAX);
125
126     BOOLEAN PROCEDURE OK(X);
127     VALUE X; REAL X;
128     BEGIN
129         IF X GTR -5.0 AND X LSS 45.0 THEN
130             BEGIN
131                 OKN:=OKN+1;
132                 OK:=TRUE;
133             END ELSE
134                 FEILN:=FEILN+1;
135         END *** OK ***;
136
137     PROCEDURE LES(FIL,UT,FE); VALUE FIL; STRING FIL;
138         LABEL UT,FE;
139     BEGIN
140         INTEGER I,H;
141         COMMENT LESINDEX PEKER PÅ FØRSTE POST I LESBUF SOM
142             SKÅL LESES;
143         IF LESINDEX GEQ PMAX THEN
144             BEGIN
145                 COMMENT LESBUF ER TØMT;
146                 READ(FILE(FIL),LESBUF,UT,UT,FE);
147                 LESINDEX:=0;
148             END;
149             H:=PLENGTH-LESINDEX;
150             FOR I:=(1,1,PLENGTH) DO
151                 POST(I):=LESBUF(I+H);
152             LESINDEX:=LESINDEX+1;
153         END *** LES ***;
154
155     PROCEDURE POSISJONER(FIL,N1,N2,UT,FE); VALUE FIL,N1,N2;
156         STRING FIL; INTEGER N1,N2; LABEL UT,FE;
157     BEGIN
158         INTEGER H;
159         H:=BMAX-PLENGTH+1;
160         LESINDEX:=0;
161     OM:  READ(FILE(FIL),LESBUF,UT,UT,FE);
162         IF LESBUF(H) LSS N1 THEN GOTO OM;
163         IF LESBUF(BMAX) LSS N2 THEN GOTO OM;
164         FOR LESINDEX:=LESINDEX+1 WHILE LESINDEX LSS PMAX AND
165             LESBUF((LESINDEX+1)*PLENGTH) LSS N2 DO;
166         END *** POSISJONER ***;
167
168     WRITE(CORE(SH1),<<X30,'*** ',S20,' FRA ',I2,'/',I1,' TIL ',I2,'/',I1,
169         I6,' ***',A1>>,SPECIES,DAG,MND,SDAG,SMND,AAR);
170     IF KORTUT THEN
171         BEGIN
172         WRITE(CARDS,<<X5,'*** ',S20,' FRA ',I2,'/',I1,' TIL ',I2,'/',I1,
173             I6,' ***',A1>>,SPECIES,DAG,MND,SDAG,SMND,AAR);
174             WRITE(CARDS,<<' TIME DAG MND AAR UTET. L.F. LYS',S10,
175                 ' EGGMID ST.AV. MIN MAX ANT.',A1>>,SH2(61,10));
176         END;
177         FOR I:=(2,1,7) DO
178             SH3(10*I+1,14):=' MIDDEL';
179             SH3(86,46):='MIDDEL ST.AVVIK MIN MAX ANTALL';
180             DATO:=100*MND+DAG;
181             SDATO:=100*SMND+SDAG;
182
183         TAKK:=DAKK:=0;
184         TANT:=DANT:=0;
185         WRITE(<<E1,S132,A4.5>>,SH1);

```

```

186 WRITE(<<S132,A1>>,SH2);
187 WRITE(<<S132,A1.2>>,SH3);
188 POSISJONER('FUGL',DATO,KL,NOTFOUND,FEIL);
189 NOK1: LES('FUGL',EOF2,FEIL);
190 IF DATO GTR POST(1) THEN GOTO NOK1;
191 TIME:=ENTIER(POST(PLENGTH)/100);
192 REFDATO:=DATO;
193 DOGN:=0;
194 TMIN:=DMIN:=45.0;
195 OM2: IF TIME EQL ENTIER(POST(PLENGTH)/100) THEN
196 BEGIN
197 FOR I:=(1,1,NF) DO
198 BEGIN
199 K:=FMPKT(I);
200 P:=POST(FINDEX(K))/100;
201 TA(K):=TA(K)+P;
202 NT(K):=NT(K)+1;
203 END;
204 FOR I:=(1,1,NE) DO
205 BEGIN
206 P:=POST(EMPKT(I))/TP;
207 IF OK(P) THEN
208 BEGIN
209 TANT:=TANT+1;
210 TAKK:=TAKK+P;
211 TKVSUM:=TKVSUM+P*P;
212 IF TMAX LSS P THEN TMAX:=P;
213 IF DMAX LSS P THEN DMAX:=P;
214 IF TMIN GTR P THEN TMIN:=P;
215 IF DMIN GTR P THEN DMIN:=P;
216 END;
217 END;
218 GOTO LES2;
219 END;
220 COMMENT TIMESKIFTE;
221 FOR I:=(1,1,NF) DO
222 BEGIN
223 K:=FMPKT(I);
224 TM(K):=IF NT(K) EQL 0 THEN 0 ELSE TA(K)/NT(K);
225 END;
226 IF TANT EQL 0 THEN TANT:=1;
227 TMID:=TAKK/TANT;
228 TVAR:=TKVSUM/TANT-TMID*TMID;
229 TVAR:=ABS(TVAR);
230 WRITE(F1,TIME,TIME+1,DAG,MND,AAR-1900,PT+TM(1),FOR I:=(2,1,6) DO
231 TM(I),TMID,SQRT(TVAR),TMIN,TMAX,TANT);
232 IF KORTUT THEN
233 WRITE(CARDS,FPUNCH,TIME,DAG,MND,AAR,PT+TM(1),FOR I:=2,4,6 DO
234 TM(I),TMID,SQRT(TVAR),TMIN,TMAX,TANT);
235 TIME:=ENTIER(POST(PLENGTH)/100);
236 DAKK:=DAKK+TAKK;
237 DKVSUM:=DKVSUM+TKVSUM;
238 DANT:=DANT+TANT;
239 TAKK:=0;
240 TKVSUM:=0; TANT:=0;
241 TMAX:=0; TMIN:=45.0;
242 FOR I:=(1,1,NF) DO
243 BEGIN
244 K:=FMPKT(I);
245 DA(K):=DA(K)+TA(K);
246 TA(K):=0;
247 ND(K):=ND(K)+NT(K);

```



```

248     NT(K):=0;
249     END;
250     IF POST(1) EQL REFDATE THEN GOTO OM2;
251     COMMENT DØGNSKIFTE;
252     WRITE(⟨⟨'DØGNVERDIER ' ,I2,' , DØGN:' ,A4.3⟩⟩,DOGN+1);
253     WRITE(⟨⟨S132,A1⟩⟩,SH2);
254     WRITE(⟨⟨S132,A1.2⟩⟩,SH3);
255     DMID:=DAKK/DANT;
256     DVAR:=DKVSUM/DANT-DMID*DMID;
257     DVAR:=ABS(DVAR);
258     FOR I:=(1,1,NF) DO
259     BEGIN
260         K:=FMPKT(I);
261         DM(K):=IF ND(K) EQL 0 THEN 0 ELSE DA(K)/ND(K);
262     END;
263     WRITE(F2,DAG,MND,AAR-1900,PT+DM(1),FOR I:=(2,1,6) DO DM(I),DMID,
264         SQRT(DVAR),DMIN,DMAX,DANT);
265     DAKK:=0;
266     DKVSUM:=0;
267     DANT:=0;
268     DMAX:=0; DMIN:=45.0;
269     FOR I:=(1,1,NF) DO
270     BEGIN
271         K:=FMPKT(I);
272         DA(K):=0;
273         ND(K):=0;
274     END;
275     DIFF:=MOD(POST(1),100)-MOD(REFDATE,100);
276     IF DIFF LSS 0 THEN DIFF:=DIFF+MND;
277     IF DIFF NEQ 1 THEN
278     BEGIN
279         I:=ENTIER(POST(1)/100);
280         K:=MOD(POST(1),100);
281         WRITE(⟨⟨'*** DATA MANGLER FOR PERIODEN ' ,I2,' / ' ,I1,' - ' ,
282             I2,' / ' ,I1,' ***' ,A2.2⟩⟩,MOD(REFDATE,100),ENTIER(REFDATE/100),
283             K,I);
284     END;
285     DOGN:=DOGN+DIFF;
286     REFDATE:=POST(1);
287     MND:=ENTIER(REFDATE/100); DAG:=REFDATE-100*MND;
288     WRITE(⟨⟨E1,S132,A4.5⟩⟩,SH1);
289     WRITE(⟨⟨S132,A1⟩⟩,SH2);
290     WRITE(⟨⟨S132,A1.2⟩⟩,SH3);
291     GOTO OM2;
292     LES2: LES('FUGL',EQF2,FEIL);
293     IF POST(1) LSS SDATO THEN GOTO OM2;
294     IF POST(PLNGTH) LSS SKL THEN GOTO OM2;
295     GOTO EQF2;
296     FEIL: WRITE('FEIL UNDER LESE/SKRIVE-OPERASJONER PÅ FILEN FUGL');
297     GOTO SLUTT;
298     EOF2:
299     FOR I:=(1,1,NF) DO
300     BEGIN
301         K:=FMPKT(I);
302         TM(K):=IF NT(K) EQL 0 THEN 0 ELSE TA(K)/NT(K);
303     END;
304     DAKK:=DAKK+TAKK;
305     DKVSUM:=DKVSUM+TKVSUM;
306     DANT:=DANT+TANT;
307     TMID:=IF TANT EQL 0 THEN 0 ELSE TAKK/TANT;
308     TVAR:=IF TANT EQL 0 THEN 0 ELSE TKVSUM/TANT-TMID*TMID;
309     TVAR:=ABS(TVAR);

```

```

310     DMID:=IF DANT EQL 0 THEN 0 ELSE DAKK/DANT;
311     DVAR:=IF DANT EQL 0 THEN 0 ELSE DKVSUM/DANT-DMID*DMID;
312     DVAR:=ABS(DVAR);
313     WRITE(F1,TIME,TIME+1,DAG,MND,AAR-1900,PT*TM(1),FOR I:=(2,1,6) DO
314         TM(I),TMID,SQRT(TVAR),TMIN,TMAX,TANT);
315     IF KORTUT THEN
316     WRITE(CARDS,FPUNCH,TIME,DAG,MND,AAR,PT*TM(1),FOR I:=2,4,6 DO
317         TM(I),TMID,SQRT(TVAR),TMIN,TMAX,TANT);
318     WRITE(<<'DØGNVERDIER ',I2,'. DØGN:',A4.3>>,DØGN+1);
319     WRITE(<<S132,A1>>,SH2);
320     WRITE(<<S132,A1.2>>,SH3);
321     FOR I:=(1,1,NF) DO
322     BEGIN
323         K:=FMPKT(I);
324         DM(K):=IF ND(K) EQL 0 THEN 0 ELSE DA(K)/ND(K);
325     END;
326     WRITE(F2,DAG,MND,AAR-1900,PT*DM(1),FOR I:=(2,1,6) DO DM(I),DMID,
327         SQRT(DVAR),DMIN,DMAX,DANT);
328     WRITE(<<D5.2,' % AV MÅLINGENE UTENFOR AKSEPTABELT INTERVALL',A5.1>>,
329         100*FEILN/(FEILN+OKN));
330     END;
331     GOTO SLUTT;
332     NOTFOUND: WRITE('FINNER IKKE DATA');
333     SLUTT: IF KORTUT THEN WRITE(<<'RESULTATENE ER PUNCHET PÅ KORT',A5.1>>);
334     END;

```

IN

