

**SENGLASIALE PALEOGEOGRAFISKE REKONSTRUKSJONSMODELLER
TESTET PÅ UTVALGTE OMRÅDER I NORGE**

**APPENDIX:
UTSKRIFT AV BEREGNINGSPROGRAMMER FOR MIKRODATAMASKIN**

Kjell Kjenstad

SAMMENDRAG

Det presenteres lister av programmer for beregning av paleogeografiske forhold og for presentasjon av resultatene på kartform. Algoritmene og oppbygningen av program-pakken er beskrevet i detalj i den avhandlingen som denne programsamlingen er appendix til. Programmene er skrevet i FORTRAN IV til bruk på en CROMEMCO-II mikrodatamaskin under operativsystemet CDOS og kartene plottes på en rastergrafisk monitor ved hjelp av styresystemet SDI.

Kjell Kjenstad, Geografisk institutt, Universitetet i Oslo,
Postboks 1042, Blindern, 0316-Oslo 3.

INNHOLDSFORTEGNELSE

GENERELL BESKRIVELSE.....	s. A - 3
PROGRAMMER SOM FORETAR ESTIMERINGER OG BEREGNINGER ETTER "SPLINE"-HAVNIVÅMODELLEN.....	s. A - 4
PROGRAMMER SOM FORETAR ESTIMERINGER OG BEREGNINGER ETTER DEN ENKLE HAVNIVÅMODELLEN.....	s. A - 20
KARTPLOTTEPROGRAMMER	s. A - 33
FORTRLIB: er subrutinebibliotek for FORTRAN subrutiner.....	s. A - 52
KARTLIB: er subrutinebibliotek for kartplottesubrutiner.....	s. A - 58

GENERELL BESKRIVELSE

Det presenteres lister av programmer for beregning av paleogeografiske forhold og for presentasjon av resultatene på kartform. Algoritmene og oppbygningen av program-pakken er beskrevet i detalj i den avhandlingen som denne programsamlingen er appendix til.

Programmene er skrevet i FORTRAN IV til bruk på en CROMEMCO-II mikrodatamaskin under operativsystemet CDOS og kartene plottes på en rastergrafisk monitor ved hjelp av styresystemet SDI.

Programmene er skrevet som selvstendige programmer. De enkelte programmene er bygd inn i tre ulike menyoverlegg for enklere behandling av programmene i et interaktivt menysystem. Det ene menyoverlegget (type 'XX') styrer alle programmer knyttet til estimering og direkteanvendelse av "spline"-havnivåmodellen, det andre menyoverlegget (type 'YY') styrer alle programmer knyttet til estimering og direkteanvendelse av den enkle havnivåmodellen, mens det tredje menyoverlegget (type 'ZZ') styrer alle kartplotteprogrammer.

Det er laget to separate biblioteker som inneholder felles subrutiner. Det ene biblioteket (FORTRLIB) er knyttet til FORTRAN-rutiner som stadig kommer tilbake, mens det andre biblioteket (KARTLIB) er knyttet til standardrutiner for kartplotting i riktig format på den grafiske hukommelsen.

Beregning av flatetilpasning av et synkront havnivå er imidlertid såpass komplisert at programmet er skrevet i FORTRAN IV for en DEC-10 datamaskin ved EDB-senteret ved Universitetet i Oslo. Dette programmet gjør bruk av matematikk-subrutinebiblioteket IMSL og parameterfilene overføres deretter til mikromaskinen for videre bruk i beregningsprogrammene.

```

*****
**
** PROGRAMMER SOM FORETAR ESTIMERING OG
** BEREGNINGER ETTER "SPLINE"-HAVNIVÅMODELLEN
**
*****
MENYPRXX: er meny-programmet for
"spline"-havnivåmodell-pakken.....s. A - 4

KURVBEXX: foretar regressjonstilpasning
etter "spline"-modellen av
strandforskyvningskurver.....s. A - 5

KURVPLXX: foretar plottning av strand-
forskyvningskurver etter "spline"
-modellen på grafisk skjerm.....s. A - 9

MODEBEXX: foretar beregning av glatt overgang
mellom strandforskyvningskurver
beregnet etter "spline"-modellen.....s. A - 12

FLATBEXX: Identisk med FLATBEYY

GRADIEXX: foretar beregning av flategradienter og
flateretninger på et sted
etter "spline"-havnivåmodell.....s. A - 15

MGCALCXX: foretar datering av MG-lokaliteter på
grunnlag av "spline"-havnivåmodellen.....s. A - 17

```

```

C*****
C
C Menyprogram for SPLINE-MODELL- programmer
C
C Programmert av Kjell Kjenstad, UiO
C Mai 1984
C
C*****
C PROGRAM MENYPR
C-----
11 CALL TDV(25)
WRITE(3,12)
12 FORMAT(1X, //
*,8X, '*****'
*,8X, '** HOVEDMENY
*,8X, '**
*,8X, '** Programmer for beregning og
*,8X, '** plottning av spline-havnivåmodeller
*,8X, '**
*,8X, '** *****
*,8X, '**
*,8X, '** 0: Slutt
*,8X, '** 1: Kurvetilpasning
*,8X, '** 2: Kurveplottning av temporærfil
*,8X, '** 3: Rom-modell-beregning
*,8X, '** 4: Beregning av flatehøyder
*,8X, '** 5: Beregning av gradienter og retninger
*,8X, '** 6: Datering av MG-lokaliteter
*,8X, '**
*/'
: ')
READ(3,13) NEXT
13 FORMAT(I2)
CALL TDV(25)
IF(NEXT.LE.0) STOP
IF(NEXT.GT.6) GOTO 11
GOTO (10,20,30,40,50,60),NEXT
C-----
10 CALL FCHAIN('KURVBEXXCOM',0)
20 CALL FCHAIN('KURVPLXXCOM',0)
30 CALL FCHAIN('MODEBEXXCOM',0)
40 CALL FCHAIN('FLATBEXXCOM',0)
50 CALL FCHAIN('GRADIEXXCOM',0)
60 CALL FCHAIN('MGCALCXXCOM',0)
C-----
END

```

```

C*****
C PROGRAM FOR TILPASNING AV SMOOTH-SPLINE-KURVER
C
C Programmet av Kjell Kjenstad, Uio
C August 1984
C
C*****
PROGRAM KURVE
  BYE FILE(11),NAVN(15),DANAVN(15)
  REAL T(20),Z(20),THJ(20),ZHU(20),DT(20),DZ(20),V(20,7)
  *,G(13,2),S(13,2),GHU(13,2),C(2,4,13),A(20,4),KURV
C MENY-----
10 CALL TDV(25)
  WRITE(3,11)
11 FORMAT(1X, /
  *,8X, '*****'
  *,8X, '***'
  *,8X, '***'
  *,8X, '***'
  *,8X, '***'
  *,8X, '*****'
  *)
C*****
C PROGRAMMODUL FOR BEREGNING AV SMOOTH-SPLINE-TILPASNING
C
C*****
C INNLESNING AV DATAFIL-----
20 CALL FREAD(FILE,'INNKURVE.DAT')
  CALL LFEED(1)
  IF(FILE(1).EQ.32) STOP
  FILE(10)='A'
  FILE(11)='T'
  CALL OPEN(6,FILE,0)
  READ(6,23,ERR=20,END=20) IBASE,X,Y,(NAVN(K),K=1,15),THL,ZHL
23 FORMAT(1I,F4.0,F5.0,1X,15A1,1X,F6.0,F4.0)
  WRITE(3,25) IBASE,X,Y,(NAVN(K),K=1,15),THL,ZHL
25 FORMAT(1X,1I,F4.0,F5.0,1X,15A1,1X,F6.0,F4.0)
  CALL LFEED(1)
  T(1)=0.
  Z(1)=0.
  I=2
DO 22 II=1,19
  READ(6,23,END=24) III,XX,YY,(DANAVN(K),K=1,15),T(I),Z(I)
  WRITE(3,25) III,XX,YY,(DANAVN(K),K=1,15),T(I),Z(I)
  Z(I)=Z(I)/ZHL
  T(I)=T(I)/1000.
  I=I+1
22 CONTINUE
24 IDAT=I-1
  IANTID=T(IDAT)
  IANTP1=IANTID+1
  ENDFILE 6
  CALL LFEED(1)
C*****
C BEREGNING AV P-VERDI FOR ENDELIG TILPASNING-----
40 DO 41 I=1,IDAT
  DZ(I)=0.01
41 CONTINUE
  CALL TWRITE(' Høyde-tilpasning',17)
  CALL PVALUE(T,Z,DZ,IDAT,20,SS,V,A,P)
  CALL RWRITE(T,Z,DZ,A,20,IDAT,ZHL)
  NEXT=I-READ(' OK ? (Ja)',12)
  IF(NEXT.EQ.0) GOTO 40
C BEREGNING AV 1000-ÅRS-PUNKTER OG GRADIENTER-----
50 CALL TWRITE(' 1000-års intervall',19)
  CALL LFEED(1)
  IPKT=1
DO 51 I=1,IANTID
  IF(IPKT.EQ.IDAT-1) GOTO 53
  IF((I.GE.T(IPKT)).AND.(I.LT.T(IPKT+1))) GOTO 53
  IPKT=IPKT+1
  GOTO 52
53 DT0=I-T(IPKT)
  G(I,1)=A(IPKT,1)+A(IPKT,2)*DT0+A(IPKT,3)*DT0*DT0/2
  +A(IPKT,4)*DT0*DT0*DT0/6
  *
  S(I,1)=A(IPKT,2)+A(IPKT,3)*DT0+A(IPKT,4)*DT0*DT0/2
  G(I,1)=G(I,1)*ZHL
  S(I,1)=S(I,1)*ZHL
  G(I,2)=0.0
  S(I,2)=0.0
  IF((G(I,1).GT.300).OR.(G(I,1).LT.-100)) S(I,1)=0.0
  IF((G(I,1).GT.300).OR.(G(I,1).LT.-100)) G(I,1)=0.0
  IF((S(I,1).GT.900).OR.(S(I,1).LT.-900)) G(I,1)=0.0
  IF((S(I,1).GT.900).OR.(S(I,1).LT.-900)) S(I,1)=0.0
  WRITE(3,54) I,G(I,1),G(I,2),S(I,1),S(I,2)
54 FORMAT(' Tid: ',I2,'000 Høyde: ',F6.1,' +- ',F6.1
  *, ' Gradient: ',F6.1,' +- ',F6.1)
  *
51 CONTINUE
  NEXT=L-READ(' OK ? (Ja)',12)
  IF(NEXT.EQ.0) GOTO 50
C ENDEGRADIENTER-----
60 ITDE=L-READ(' Gi deglasijsjonstidspunkt: ',29)
  IF(ITDE.EQ.0) ITDE=T(IDAT)
  IZDE=L-READ(' Gi deglasijsjonshøyde: ',25)
  IF(IZDE.EQ.0) IZDE=Z(IDAT)
  IGDE=L-READ(' Gi deglasijsjonsgradient: ',28)
  IF(IGDE.EQ.0) IGDE=ZHL/2
  T0=ITDE/1000.-IANTID
  D1=G(IANTID,1)
  D2=S(IANTID,1)
  D3=IZDE
  D4=IGDE
  X1=D1
  X2=D2
  X3=(-(D2+D4)/T0)+(3*(D3-D1)/(T0*T0))
  X4=((D2+D4)/(T0*T0)+(2*(D1-D3)/(T0*T0*T0))
  G(IANTP1,1)=X1+X2+X3+X4
  S(IANTP1,1)=X2+2*X3+3*X4
  G(IANTP1,2)=0.0
  S(IANTP1,2)=0.0

```

```

WRITE(3,54)
* IANTP1,G(IANTP1,1),G(IANTP1,2),S(IANTP1,1),S(IANTP1,2)
NEXT=LREAD(' OK ? (Ja)',12)
IF(NEXT.EQ.0) GOTO 60
C BEREGNING AV KUBISK INTERPOLATOR-----
70 DO 71 I=1,IANTID
C(1,1,I)=G(I,1)/ZHL
C(1,2,I)=S(I,1)/ZHL
C(1,3,I)=(-3*G(I,1)+3*G(I+1,1)-2*S(I,1)-S(I+1,1))/ZHL
C(1,4,I)=(2*G(I,1)-2*G(I+1,1)+S(I,1)+S(I+1,1))/ZHL
71 CONTINUE
IF(IANTP1.GT.13) GOTO 60
DO 72 I=IANTP1,13
C(1,1,I)=C(1,1,I-1)+C(1,2,I-1)+C(1,3,I-1)+C(1,4,I-1)
C(1,2,I)=C(1,2,I-1)+2*C(1,3,I-1)+3*C(1,4,I-1)
C(1,3,I)=C(1,3,I-1)+3*C(1,4,I-1)
C(1,4,I)=C(1,4,I-1)
72 CONTINUE
C UTSKRIFT AV RESIDUALER-----
80 CALL TWRITE(' Endelige residualer',20)
SD=0.0
WRITE(3,82)
82 FORMAT(' Nr. Tid Høyde Kurve Usikk. Resid. Grad. ')
DO 81 I=2,IDAT
TID0=T(I)*1000.
HOYD=Z(I)*ZHL
DHOY=DZ(I)*ZHL
IT0=T(I)
DT0=T(I)-IT0
KURV=(C(1,1,IT0)+C(1,2,IT0)*DT0+C(1,3,IT0)*DT0*DT0/2
+C(1,4,IT0)*DT0*DT0*DT0/6)*ZHL
RESI=HOYD-KURV
GRAD=(C(1,2,IT0)+C(1,3,IT0)*DT0+C(1,4,IT0)*DT0*DT0/2)*ZHL
WRITE(3,85) I,TID0,HOYD,KURV,DHOY,RESI,GRAD
85 FORMAT(1X,I2,F7.0,5F8.2)
SD=SD+RESI*RESI
81 CONTINUE
SD=SD+RESI*RESI
DO 83 I=2,IDAT
DZ(I)=SD*2./ZHL
83 CONTINUE
WRITE(3,84) SD
84 FORMAT(' Standardavvik på z:',F6.2)
NEXT=LREAD(' OK ? (Ja)',12)
IF(NEXT.EQ.0) GOTO 40
C*****
C
C PROGRAMMODUL FOR BEREGNING AV USIKKERHETS - TILPASNING
C*****
100 Z(1)=0.
T(1)=0.
DZ(1)=0.
SUM=0.0
DO 101 I=2,IDAT
IT0=T(I)
DT0=T(I)-IT0
HOYD=C(1,1,IT0)+C(1,2,IT0)*DT0+C(1,3,IT0)*DT0*DT0/2
+C(1,4,IT0)*DT0*DT0*DT0/6
Z(I)=ABS(Z(I)-HOYD)
SUM=SUM+Z(I)
101 CONTINUE
SUM=SUM/IDAT
DO 103 I=2,IDAT
DZ(I)=SUM/4.
103 CONTINUE
C BEREGNING AV P-VERDI-----
105 CALL TWRITE(' Usikkerhets-tilpasning',23)
CALL PVALUE(T,Z,DZ,IDAT,20,SS,V,A,P)
CALL RWRITE(T,Z,DZ,A,20,IDAT,ZHL)
NEXT=LREAD(' OK ? (Ja)',12)
IF(NEXT.EQ.0) GOTO 105
C BEREGNING AV 1000-ÅRS-PUNKTER OG GRADIENTER-----
110 IPKT=1
DZHL=ZHL/10.
DO 111 I=1,IANTID
IF(IPKT.EQ.IDAT-1) GOTO 113
IF((I.GE.T(IPKT)).AND.(I.LT.T(IPKT+1))) GOTO 113
IPKT=IPKT+1
GOTO 112
113 DT0=I-T(IPKT)
G(I,2)=A(IPKT,1)+A(IPKT,2)*DT0+A(IPKT,3)*DT0*DT0/2
+A(IPKT,4)*DT0*DT0*DT0/6
S(I,2)=A(IPKT,2)+A(IPKT,3)*DT0+A(IPKT,4)*DT0*DT0/2
G(I,2)=G(I,2)*ZHL
S(I,2)=S(I,2)*ZHL
IF(G(I,2).GT.DZHL) G(I,2)=DZHL
IF(G(I,2).LT.0.) G(I,2)=0.0
IF(S(I,2).GT.DZHL) S(I,2)=DZHL
IF(S(I,2).LT.-DZHL) S(I,2)=-DZHL
IF(S(IANTID,2).LT.0.0) S(IANTID,2)=0.0
WRITE(3,54) I,G(I,1),G(I,2),S(I,1),S(I,2)
111 CONTINUE
C ENDEGRADIENTER-----
120 IZDE=DZHL
IGDE=DZHL
T0=ITDE/1000.-IANTID
D1=G(IANTID,2)
D2=S(IANTID,2)
D3=IZDE
D4=IGDE
X1=D1
X2=D2
X3=(-(2*D2+D4)/T0)+(3*(D3-D1)/(T0*T0))
X4=((D2+D4)/(T0*T0))+(2*(D1-D3)/(T0*T0*T0))
G(IANTP1,2)=X1+X2+X3+X4
S(IANTP1,2)=X2+2*X3+3*X4
WRITE(3,54)
*IANTP1,G(IANTP1,1),G(IANTP1,2),S(IANTP1,1),S(IANTP1,2)
C BEREGNING AV KUBISK INTERPOLATOR-----
DO 121 I=1,IANTID
C(2,1,I)=G(I,2)/ZHL

```

```

C(2,2,I)=S(I,2)/ZHL
C(2,3,I)=(-3*G(I,2)+3*G(I+1,2)-2*S(I,2)-S(I+1,2))/ZHL
C(2,4,I)=(2*G(I,2)-2*G(I+1,2)+S(I,2)+S(I+1,2))/ZHL
121 CONTINUE
IF(IANTP1.GT.13) GOTO 200
DO 122 I=IANTP1,13
  C(2,1,I)=C(2,1,I-1)+C(2,2,I-1)+C(2,3,I-1)+C(2,4,I-1)
  C(2,2,I)=C(2,2,I-1)+2*C(2,3,I-1)+3*C(2,4,I-1)
  C(2,3,I)=C(2,3,I-1)+3*C(2,4,I-1)
  C(2,4,I)=C(2,4,I-1)
122 CONTINUE
NEXT=LREAD(' OK ? (Ja)',12)
CALL LFEED(1)
IF(NEXT) 200,105,200
C*****
C UTSKRIFT AV KOEFFISIENTFIL
C*****
200 FILE(9)='S'
FILE(10)='P'
FILE(11)='L'
CALL OPEN(7,FILE,0)
WRITE(7,25) IBASE,X,Y,(NAVN(K),K=1,15),THL,ZHL
WRITE(3,25) IBASE,X,Y,(NAVN(K),K=1,15),THL,ZHL
DO 201 I=1,13
  WRITE(7,202) I,((C(J,K,I),K=1,4),J=1,2)
202 FORMAT(1X,12,1X,8E15.9)
201 CONTINUE
ENDFILE 7
C SLUTT-----
CALL FCHAIN('KURVPLXXCOM',0)
END
C*****
C SUBROUTINE FOR LETING ETTER RIKTIG P-VERDI
C*****
SUBROUTINE PVALUE(X,Y,DY,NPOINT,NDIM,S,V,A,P)
REAL A(NDIM,4),DY(NDIM),V(NDIM,7),X(NDIM),Y(NDIM)
CALL LFEED(1)
P=JREAD(' Gi p-%-verdi (p=0: glatt, p=100: krum) :',41)/100.
IF(P.GE.1.) P=0.998
IF(P.LE.0.) P=0.002
30 EMIN=-35.
SMIN=10.**EMIN
SFP=SMOOTH(X,Y,DY,NPOINT,NDIM,SMIN,V,A,PMIN)
EMAX=35.
SMAX=10.**EMAX
SFP=SMOOTH(X,Y,DY,NPOINT,NDIM,SMAX,V,A,PMAX)
31 DO 32 I=1,50
  EMID=(EMIN+EMAX)/2.
  SMID=10.**EMID
  SFP=SMOOTH(X,Y,DY,NPOINT,NDIM,SMID,V,A,PMID)
  IF((ABS(PMID-P)).LT.0.001) GOTO 35
  IF((P.LE.PMAX).AND.(P.GT.PMID)) GOTO 33

```

```

IF((P.LE.PMIN).AND.(P.GT.PMID)) GOTO 34
  EMIN=EMID
  SMIN=SMID
  IF((PMIN.EQ.PMID).AND.(PMID.NE.1.).AND.(PMID.NE.0.)) GOTO 35
  PMIN=PMID
GOTO 32
34 EMAX=EMID
  SMAX=SMID
  IF((PMAX.EQ.PMID).AND.(PMID.NE.1.).AND.(PMID.NE.0.)) GOTO 35
  PMAX=PMID
32 CONTINUE
IF(PMID.GE.1.0) SFP=SMOOTH(X,Y,DY,NPOINT,NDIM,SMAX,V,A,PMID)
35 WRITE(3,39) I,SMID,PMID
39 FORMAT(' Antall iterasjoner:',I2,' S=',E8.2,' P=',F7.5/)
RETURN
END
C*****
C SUBROUTINE FOR SKRIVING AV DELRESULTATER
C*****
SUBROUTINE RWRITE(X,Y,DY,A,NDIM,IDAT,ZHL)
REAL A(NDIM,4),X(NDIM),Y(NDIM),DY(NDIM),KURV
SD=0.0
WRITE(3,30)
30 FORMAT(' Nr. Tid Høyde Kurve Usikk. Resid. Grad. ')
DO 31 I=2,IDAT
  TIDO=X(I)*1000.
  HOYD=Y(I)*ZHL
  DHOY=DY(I)*ZHL
  KURV=A(I,1)*ZHL
  RESI=HOYD-KURV
  GRAD=A(I,2)*ZHL
  IF((GRAD.GT.9999.0).OR.(GRAD.LT.-9999.0)) GRAD=9999.0
  WRITE(3,32) I,TIDO,HOYD,KURV,DHOY,RESI,GRAD
32 FORMAT(1X,I2,F7.0,5F8.2)
  SD=SD+RESI*RESI
31 CONTINUE
SD=SQRT(SD/IDAT)
WRITE(3,33) SD
33 FORMAT('/ Standardavvik på z:',F6.2)
RETURN
END
C*****
C SUBROUTINE FOR GLATTING AV DATAPUNKTER
C*****
FUNCTION SMOOTH(X,Y,DY,NPOINT,NDIM,S,V,A,P)
REAL A(NDIM,4),DY(NDIM),V(NDIM,7),X(NDIM),Y(NDIM)
CALL SETUPQ(X,DY,Y,NPOINT,NDIM,V,A(1,4))
IF(S.GT.0) GOTO 20
10 P=1
CALL CHOLID(P,V,A(1,4),NPOINT,NDIM,1,A(1,3),A(1,1))
SFP=0.

```

```

C
GOTO 60
20 P=0
CALL CHOLD(P,V,A(1,4),NPOINT,NDIM,1,A(1,3),A(1,1))
SFP=0.
DO 21 I=1,NPOINT
SFP=SFP+(A(I,1)*DY(I))**2
21 CONTINUE
SFP=SFP*36.
IF(SFP.LE.S) GOTO 60
PREVSF=SFP
UTRU=0.
DO 25 I=2,NPOINT
UTRU=UTRU+V(I-1,4)*(A(I-1,3)*(A(I-1,3)+A(I,3))+A(I,3)**2)
25 CONTINUE
P=(SFP-S)/(24.*UTRU)
C
30 CALL CHOLD(P,V,A(1,4),NPOINT,NDIM,1,A(1,3),A(1,1))
SFP=0.
DO 35 I=1,NPOINT
SFP=SFP+(A(I,1)*DY(I))**2
35 CONTINUE
SFP=SFP*36.*(1.-P)**2
IF(SFP.LE.1.01*S) GOTO 60
IF(SFP.GE.PREVSF) GOTO 10
CHANGE=(P-PREVP)/(SFP-PREVSF)*(SFP-S)
PREVP=P
P=P-CHANGE
PREVSF=SFP
IF(P.LT.1.) GOTO 30
P=1.-SQRT(S/PREVSF)*(1.-PREVP)
GOTO 30
C
60 SMOOTH=SFP
SIXIMP=6.*(1.-P)
DO 61 I=1,NPOINT
A(I,1)=Y(I)-SIXIMP*DY(I)**2*A(I,1)
61 CONTINUE
SIXP=6.*P
DO 62 I=1,NPOINT
A(I,3)=A(I,3)*SIXP
62 CONTINUE
NPM1=NPOINT-1
DO 63 I=1,NPM1
A(I,4)=(A(I+1,3)-A(I,3))/V(I,4)
A(I,2)=(A(I+1,1)-A(I,1))/V(I,4)
* -(A(I,3)+A(I,4)/3.*V(I,4))/2.*V(I,4)
63 CONTINUE
RETURN
END
C*****
C
C SUBROUTINE FOR OPSETTING AV ARRAYS
C*****
SUBROUTINE CHOLD(P,V,PTY,NPOINT,NDIM,NCOL,U,QU)
REAL QTY(NDIM),QU(NDIM),U(NDIM),V(NDIM),V(NDIM)
NPM1=NPOINT-1
DO 2 I=2,NPM1
V(I,1)=SIXIMP*V(I,5)+TWO*P*(V(I-1,4)+V(I,4))
V(I,2)=SIXIMP*V(I,6)+P*V(I,4)
V(I,3)=SIXIMP*V(I,7)
SIXIMP=6.*(1.-P)
TWO*P=2.*P
DO 2 I=2,NPM1
V(I,1)=SIXIMP*V(I,5)+TWO*P*(V(I-1,4)+V(I,4))
V(I,2)=SIXIMP*V(I,6)+P*V(I,4)
V(I,3)=SIXIMP*V(I,7)
2 CONTINUE
NPM2=NPOINT-2
IF(NPM2.GE.2) GOTO 10
U(1)=0.
U(2)=QTY(2)/V(2,1)
U(3)=0.
GOTO 41
C*****
C
C SUBROUTINE SETUPO(X,DX,Y,NPOINT,NDIM,V,PTY)
REAL DX(NDIM),QTY(NDIM),V(NDIM,7),X(NDIM),Y(NDIM)
NPM1=NPOINT-1
DO 11 I=2,NPM1
V(I,4)=X(I+1)-X(I)
V(I,1)=DX(I-1)/V(I-1,4)
V(I,2)=-DX(I)/V(I,4)-DX(I)/V(I-1,4)
V(I,3)=DX(I+1)/V(I,4)
11 CONTINUE
V(NPOINT,1)=0.
DO 12 I=2,NPM1
V(I,5)=V(I,1)**2+V(I,2)**2+V(I,3)**2
12 CONTINUE
IF(NPM1.LT.3) GOTO 14
DO 13 I=3,NPM1
V(I-1,6)=V(I-1,2)*V(I,1)+V(I-1,3)*V(I,2)
13 CONTINUE
14 V(NPM1,6)=0.
IF(NPM1.LT.4) GOTO 16
DO 15 I=4,NPM1
V(I-2,7)=V(I-2,3)*V(I,1)
15 CONTINUE
16 V(NPM1-1,7)=0.
V(NPM1,7)=0.
C
PREV=(Y(2)-Y(1))/V(1,4)
DO 21 I=2,NPM1
DIFF=(Y(I+1)-Y(I))/V(I,4)
QTY(I)=DIFF-PREV
PREV=DIFF
21 CONTINUE
RETURN
END
C*****
C
C SUBROUTINE FOR OPSETTING AV ARRAYS
C*****
SUBROUTINE CHOLD(P,V,PTY,NPOINT,NDIM,NCOL,U,QU)
REAL QTY(NDIM),QU(NDIM),U(NDIM),V(NDIM),V(NDIM)
NPM1=NPOINT-1
DO 2 I=2,NPM1
V(I,1)=SIXIMP*V(I,5)+TWO*P*(V(I-1,4)+V(I,4))
V(I,2)=SIXIMP*V(I,6)+P*V(I,4)
V(I,3)=SIXIMP*V(I,7)
SIXIMP=6.*(1.-P)
TWO*P=2.*P
DO 2 I=2,NPM1
V(I,1)=SIXIMP*V(I,5)+TWO*P*(V(I-1,4)+V(I,4))
V(I,2)=SIXIMP*V(I,6)+P*V(I,4)
V(I,3)=SIXIMP*V(I,7)
2 CONTINUE
NPM2=NPOINT-2
IF(NPM2.GE.2) GOTO 10
U(1)=0.
U(2)=QTY(2)/V(2,1)
U(3)=0.
GOTO 41
C*****
C

```



```
10 DO 20 I=2,NPM2
    RATIO=V(I,2)/V(I,1)
    V(I+1,1)=V(I+1,1)-RATIO*V(I,2)
    V(I+1,2)=V(I+1,2)-RATIO*V(I,3)
    V(I,2)=RATIO
    RATIO=V(I,3)/V(I,1)
    V(I+2,1)=V(I+2,1)-RATIO*V(I,3)
    V(I,3)=RATIO
20 CONTINUE
C
    U(1)=0.
    V(1,3)=0.
    U(2)=QTY(2)
    DO 30 I=2,NPM2
        U(I+1)=QTY(I+1)-V(I,2)*U(I)-V(I-1,3)*U(I-1)
30 CONTINUE
C
    U(NPOINT)=0.
    U(NPM1)=U(NPM1)/V(NPM1,1)
    M1=-1
    DO 40 I=NPM2,2,M1
        U(I)=U(I)/V(I,1)-U(I+1)*V(I,2)-U(I+2)*V(I,3)
40 CONTINUE
C
    41 PREV=0.
    DO 50 I=2,NPOINT
        QU(I)=(U(I)-U(I-1))/V(I-1,4)
        QU(I-1)=QU(I)-PREV
        PREV=QU(I)
50 CONTINUE
    QU(NPOINT)=-QU(NPOINT)
    RETURN
    END

C *****
C PROGRAM FOR PLOTTING AV SPLINE-KURVER
C
C Programmert av Kjell Kjenstad, Uio
C Mai 1984
C *****
PROGRAM SPLIPL
BYTE FILE(11),NAVN(15),DANAVN(15)
REAL T(20),Z(20),C1(4,13),C2(4,13)
INTEGER IVEKT(13)
C MENY-----
10 CALL TDV(25)
WRITE(3,11)
11 FORMAT(1X, /
*,8X, '*****'
*,8X, '***'
*,8X, '***'
*,8X, '***'
*,8X, '***'
*,8X, '*****'
*)
DO 12 I=1,4
DO 12 J=1,13
C1(I,J)=0.0
C2(I,J)=0.0
12 CONTINUE
C INNLESNING AV DATAFIL-----
20 CALL FIREAD(FILE,'INNKURVE.DAT')
CALL LFEED(1)
IF(FILE(1).EQ.32) STOP
IPLO=LREAD(' Plotting av dateringer (Ja)',28)
IF(IPLO) 40,40,21
21 FILE(9)='D'
FILE(10)='A'
FILE(11)='T'
CALL OPEN(6,FILE,0)
READ(6,23,ERR=20,END=20) X,Y,(NAVN(K),K=1,15),THL,ZHL
I=1
DO 22 II=1,20
READ(6,23,END=24) XX,YY,(DANAVN(K),K=1,15),T(I),Z(I)
I=I+1
22 CONTINUE
24 IDAT=I-1
ENDFILE 6
C INNLESNING AV Koeffisientfil-----
40 FILE(9)='S'
FILE(10)='P'
FILE(11)='L'
CALL OPEN(7,FILE,0)
READ(7,23,ERR=20,END=20) X,Y,(NAVN(K),K=1,15),THL,ZHL
WRITE(3,25) X,Y,(NAVN(K),K=1,15),THL,ZHL
25 FORMAT(1X,2F5.0,1X,15A1,1X,F6.0,F4.0)
--
```



```

* ,8X, '**      2: Nytt diagram
* ,8X, '**
* ,8X, '*****
*////)
NEXT=IREAD(' :',10)
CALL TDV(25)
IF(NEXT-1) 99,50,10
C SLUTT-----
99 CALL FCHAIN('MENYPRXXCOM',0)
END
C SUBROUTINE DIAGRAMME*****
SUBROUTINE RAMME(JMG,JMAX,NAVN)
  BYTE NAVN(15),TONAVN(46)
  DATA TONAVN/'S','H','O','R','E',' ','D','I','S','P','L','A',
*'E','M','E','N','T',' ','C','U','R','V','E',' ','F','R',
*'O','M',' ',' ',' ',' ',' ',' ',' ',' ',' ',' ',' ',' ',' ','
* ',' ','
  TONAVN(46)='^'
  DO 21 I=31,45
    TONAVN(I)=NAVN(I-30)
  21 CONTINUE
  CALL HXTEXT(200,440,1,TONAVN)
  CALL HXTEXT(87,70,1,13^')
  CALL HXTEXT(127,70,1,12^')
  CALL HXTEXT(167,70,1,11^')
  CALL HXTEXT(207,70,1,10^')
  CALL HXTEXT(257,70,1,9^')
  CALL HXTEXT(297,70,1,8^')
  CALL HXTEXT(337,70,1,7^')
  CALL HXTEXT(377,70,1,6^')
  CALL HXTEXT(417,70,1,5^')
  CALL HXTEXT(457,70,1,4^')
  CALL HXTEXT(497,70,1,3^')
  CALL HXTEXT(537,70,1,2^')
  CALL HXTEXT(577,70,1,1^')
  CALL HXTEXT(617,70,1,0^')
  CALL HXTEXT(500,40,1,'1000 YEARS BP^')
C-----
DO 31 I=0,13
  II=2*I*20+100
  CALL HXLINE(II,94,II,380,1)
31 CONTINUE
C-----
DO 32 J=0,14
  JJ=2*J*10+100
  CALL HXLINE(100,JJ,626,JJ,1)
32 CONTINUE
C-----
IF(JMG.GT.120) JMAX=240
IF((JMG.GT.60).AND.(JMG.LE.120)) JMAX=120
IF((JMG.GT.30).AND.(JMG.LE.60)) JMAX=60
IF(JMG.LE.30) JMAX=30
JMAX=JMAX/30
JMAX=JMAX*30
IM=JMAX/30
IF(IM.EQ.4) IM=3

```

```

IF(IM.EQ.8) IM=4
CALL HXTEXT(630,136,1,' 0 M^')
GOTO (300,310,320,330),IM
300 CALL HXTEXT(630,96,1,' -5 M^')
CALL HXTEXT(630,176,1,' 5 M^')
CALL HXTEXT(630,216,1,' 10 M^')
CALL HXTEXT(630,256,1,' 15 M^')
CALL HXTEXT(630,296,1,' 20 M^')
CALL HXTEXT(630,336,1,' 25 M^')
CALL HXTEXT(630,376,1,' 30 M^')
GOTO 350
310 CALL HXTEXT(630,96,1,' -10 M^')
CALL HXTEXT(630,176,1,' 10 M^')
CALL HXTEXT(630,216,1,' 20 M^')
CALL HXTEXT(630,256,1,' 30 M^')
CALL HXTEXT(630,296,1,' 40 M^')
CALL HXTEXT(630,336,1,' 50 M^')
CALL HXTEXT(630,376,1,' 60 M^')
GOTO 350
320 CALL HXTEXT(630,96,1,' -20 M^')
CALL HXTEXT(630,176,1,' 20 M^')
CALL HXTEXT(630,216,1,' 40 M^')
CALL HXTEXT(630,256,1,' 60 M^')
CALL HXTEXT(630,296,1,' 80 M^')
CALL HXTEXT(630,336,1,' 100 M^')
CALL HXTEXT(630,376,1,' 120 M^')
GOTO 350
330 CALL HXTEXT(630,96,1,' -40 M^')
CALL HXTEXT(630,176,1,' 40 M^')
CALL HXTEXT(630,216,1,' 80 M^')
CALL HXTEXT(630,256,1,' 120 M^')
CALL HXTEXT(630,296,1,' 160 M^')
CALL HXTEXT(630,336,1,' 200 M^')
CALL HXTEXT(630,376,1,' 240 M^')
350 RETURN
END

```

```

C*****
C
C PROGRAM FOR BEREGNING AV GLATT OVERGANG MELLOM SPLINE-KURVER
C
C Programmet av Kjell Kjenstad, UiO
C Mai 1984
C
C*****
PROGRAM MODEBE
  BYTE HFILE(11),FILE(11),NAVN(15),DANAVN(15)
  REAL SARR(10,10),X(10),Y(10),SINV(10,10),ZVEKT(10),SVEKT(10)
  *,DVEKT(7),H(5,5),CC1(4,13,10),CC2(4,13,10),C1(4,13),C2(4,13)
  DOUBLE PRECISION AAASUM,BBSUM,AAAHOF,AAADIF,BBB,CCC,DDD
  INTEGER IVEKT(10)
  DATA HFILE/'H','L',2*'X',4*' ','M','A','T' /
C-----
10 CALL TDV(25)
  WRITE(3,11)
11 FORMAT(1X, //
  *,8X, '*****' //
  *,8X, '***' //
  *,8X, '***' //
  *,8X, '***' //
  *,8X, '***' //
  *,8X, '***' //
  *,8X, '***' //
  *,8X, '***' //
  *,8X, '***' //
  *,8X, '***' //
  *,8X, '***' //
  *,8X, '*****' //
  )
  Program for beregning av
  tid-rom-modell for flere
  spline-tilpassede strand-
  forskyvningskurver
C
C NULLSTILLING-----
DO 13 I=1,4
  DO 13 J=1,13
    C1(I,J)=0.0
    C2(I,J)=0.0
    DO 13 K=1,10
      CC1(I,J,K)=0.0
      CC2(I,J,K)=0.0
13 CONTINUE
C INNLESNING AV SPLINE-PARAMETERFILER-----
20 IORD=4
IKURV=IREAD(' Gi antallet kurver som skal med i modellen: ',45)
IF((IKURV.LE.0).OR.(IKURV.GT.10)) CALL FCHAIN('MENYPRXXCOM',0)
CALL TWRITE(' Gi navnet på datafilene: ',30)
CALL LFEED(1)
DO 21 II=1,IKURV
  CALL FREAD(FILE,'INNKURVE.DAT')
  CALL LFEED(1)
  IF(FILE(1).EQ.32) GOTO 10
  FILE(9)='S'
  FILE(10)='P'
  FILE(11)='L'
  CALL OPEN(8,FILE,0)
  READ(8,25,FRR=10) IBASE,X(II),Y(II),(NAVN(K),K=1,15),THL,ZHL
  FORMAT(11,F4.0,F5.0,1X,15A1,1X,F6.0,F4.0)
  WRITE(3,26) IBASE,X(II),Y(II),(NAVN(K),K=1,15),THL,ZHL
  FORMAT(1X,11,F4.0,F5.0,1X,15A1,1X,F6.0,F4.0)
25
26
C
C*****
X(II)=X(II)/1000.
Y(II)=(Y(II)-6000.)/1000.
DO 22 I=1,13
  READ(8,23,END=24) I,(CC1(K,I,II),K=1,4),(CC2(K,I,II),K=1,4)
23 FORMAT(1X,I2,1X,8E15.9)
24 CONTINUE
25 ENDFILE 8
26 CALL LFEED(1)
21 CONTINUE
C INNLESNING AV HOVEDPLATE-----
HFILE(5)=48+IBASE
CALL OPEN(7,HFILE,0)
READ(7,15) IHORD
15 FORMAT(I15)
IH=IHORD+1
DO 16 I=1,IH
  READ(7,17) (H(I,J),J=1,IH)
17 FORMAT(6F15.9)
16 CONTINUE
ENDFILE 7
C INVERTERING AV STEDMATRISE-----
30 DO 31 I=1,10
  DO 31 J=1,10
    SINV(I,J)=0.0
31 CONTINUE
IF( (IKURV-2) 32,33,34
C 1 KURVE-----
32 SINV(1,1)=1.0
  IKURV1=1
  GOTO 40
C 2 KURVER-----
33 FKONST=(X(1)-X(2))**2+(Y(1)-Y(2))**2
  SINV(1,1)=(X(2)-X(1))*X(2)+(Y(2)-Y(1))*Y(2)/FKONST
  SINV(1,2)=(X(1)-X(2))*X(1)+(Y(1)-Y(2))*Y(1)/FKONST
  SINV(1,3)=0.0
  SINV(2,1)=(X(1)-X(2))/FKONST
  SINV(2,2)=(X(2)-X(1))/FKONST
  SINV(2,3)=0.0
  SINV(3,1)=(Y(1)-Y(2))/FKONST
  SINV(3,2)=(Y(2)-Y(1))/FKONST
  SINV(3,3)=0.0
  IKURV1=3
  GOTO 40
C FLERE KURVER-----
34 DO 35 I=1,IKURV
  SARR(I,1)=1.
  SARR(I,2)=X(I)
  SARR(I,3)=Y(I)
  SARR(I,4)=X(I)*X(I)
  SARR(I,5)=X(I)*Y(I)
  SARR(I,6)=Y(I)*Y(I)
  SARR(I,7)=X(I)*X(I)*X(I)
  SARR(I,8)=X(I)*X(I)*Y(I)
  SARR(I,9)=X(I)*Y(I)*Y(I)
  SARR(I,10)=Y(I)*Y(I)*Y(I)
35 CONTINUE

```



```

80 HOY=IREAD(' Gi høyde: ',11)
IF(IHOY.LE.0.0) GOTO 67
HOYD=HOY/ZZHL
USIKK=IREAD(' Gi usikkerhet i høydemåling: ',30)
USIKK=USIKK/ZZHL
TIDMIN=1000.
TIDO=TCAL(HOYD,TIDMIN,C1)
DHOYD=HCAL(TIDO,C2)
TID1=TCAL((HOYD-DHOYD-DZHL-USIKK),(TIDMIN-1000),C1)
TID2=TCAL((HOYD+DHOYD+DZHL+USIKK),TIDMIN,C1)
TID1=ABS(TIDO-TID1)
TID2=ABS(TID2-TIDO)
TIDM=(TID1+TID2)/2.
IT=TIDO/10
TIDO=10*IT
IT=TIDM/10
TIDM=10*IT
IF(TIDM.GE.1000.) TIDM=999
IF(TIDM.LE.-100.) TIDM=999
WRITE(3,82) HOY,TID0,TIDM
82 FORMAT(' Høyde: ',I3,' Tid: ',F6.0,'+-',F4.0//)
GOTO 80
C UTSKRIFT AV C-KORREKTISJENTER-----
90 XKOORD=1000.*XKOORD
YKOORD=1000.*YKOORD+6000.
TTHL=10600.
DO 93 I=1,8
FILE(I)=NAVN(I)
93 CONTINUE
FILE(9)='S'
FILE(10)='P'
FILE(11)='L'
CALL OPEN(9,FILE,0)
WRITE(9,94) IBASE,XKOORD,YKOORD,(NAVN(K),K=1,15),TTHL,ZHL
94 FORMAT(1X,I1,F4.0,F5.0,1X,15A1,1X,F6.0,F4.0)
DO 95 I=1,13
IF(C1(1,I).EQ.0.0) GOTO 98
WRITE(3,96) I,(C1(K,I),K=1,4),(C2(K,I),K=1,4)
96 FORMAT(' Tid: ',I2,'000',5X,2(4F6.3,5X))
WRITE(9,97) I,(C1(K,I),K=1,4),(C2(K,I),K=1,4)
97 FORMAT(1X,I2,1X,8E15.9)
95 CONTINUE
98 WRITE(3,94) IBASE,XKOORD,YKOORD,(NAVN(K),K=1,15),TTHL,ZHL
ENDFILE 9
GOTO 67
C SLUTT-----
END
C*****
C
C BEREKNING AV TIDSPUNKT I HAVMODELL
C
C*****
FUNCTION TCAL(HOYD,TIDMIN,C1)
REAL C1(4,13)
IMIN=TIDMIN
DO 21 I=IMIN,14000,1000

```

```

TID1=I-1000
TID2=I
HOYD1=HCAL(TID1,C1)
HOYD2=HCAL(TID2,C1)
IF(((HOYD.GE.HOYD2).AND.(HOYD.GE.HOYD1))
*.OR.((HOYD.LE.HOYD2).AND.(HOYD.LE.HOYD1))) GOTO 21
DO 22 J=1,7
TIDM=(TID1+TID2)/2
HOYD1=HCAL(TID1,C1)
HOYD2=HCAL(TID2,C1)
HOYDM=HCAL(TIDM,C1)
IF(((HOYD.LE.HOYD2).AND.(HOYD.GT.HOYDM))
*.OR.((HOYD.GE.HOYD2).AND.(HOYD.LT.HOYDM))) GOTO 23
IF(((HOYD.LE.HOYD1).AND.(HOYD.GT.HOYDM))
*.OR.((HOYD.GE.HOYD1).AND.(HOYD.LT.HOYDM))) GOTO 24
23 TID1=TIDM
GOTO 22
24 TID2=TIDM
GOTO 22
22 CONTINUE
TCAL=TIDM
RETURN
21 CONTINUE
TCAL=14000
RETURN
END
C*****
C
C HØYDEBEREGNINGSFUNKSJON
C
C*****
FUNCTION HCAL(TID,C1)
REAL C1(4,13)
ITID=TID/1000.
DT=TID/1000.-ITID
HCAL=
*C1(1,ITID)+C1(2,ITID)*DT+C1(3,ITID)*DT+C1(4,ITID)*DT*DT*DT
RETURN
END

```

```

C*****
C
C Program for gradientberegninger på havnivåflater i spline-modell
C
C Programmet av Kjell Kjenstad, Uio
C Juni 1984
C
C*****
PROGRAM GRADIE
BYTE HFILE(11),FILE(11),NAVN(15),DANAVN(15)
REAL SARR(10,10),X(10),Y(10),SINV(10,10),ZVEKT(10),SVEKT(10)
*,DVEKT(7),H(5,5),CC1(4,13,10),CI(4,13,4),Z(2,2)
DOUBLE PRECISION AAASUM,BBSUM,AAAHOF,AAADIF,BBB,CCC,DDD
INTEGER IVEKT(10)
DATA HFILE/'H','L',2*'X',4*' ','M','A','T' /
C-----
10 CALL TDV(25)
WRITE(3,11)
11 FORMAT(1X, /
*,8X,'*****',
*,8X,'**
*,8X,'** Program for beregning av gradienter i modell basert
*,8X,'** på flere spline-tilpassede strandforskningskurver
*,8X,'**
*,8X,'*****',
C NULLSTILLING-----
DO 13 I=1,4
DO 13 J=1,13
DO 14 K=1,4
CI(I,J,K)=0.0
14 CONTINUE
DO 13 K=1,10
CC1(I,J,K)=0.0
13 CONTINUE
C INNLESNING AV SPLINE-PARAMETERFILER-----
20 IORD=4
IKURV=IREAD(' Gi antallet kurver som skal med i modellen: ',45)
IF((IKURV.LE.0).OR.(IKURV.GT.10)) CALL FCHAIN('MENYPRXXCOM',0)
CALL TWRITE(' Gi navnet på datafilene: ',30)
CALL LFEED(1)
DO 21 II=1,IKURV
CALL FREAD(FILE,'INNKURVE.DAT')
CALL LFEED(1)
IF(FILE(1).EQ.32) GOTO 10
FILE(9)='S'
FILE(10)='P'
FILE(11)='L'
CALL OPEN(8,FILE,0)
READ(8,25,FRR=10) IBASE,X(II),Y(II),(NAVN(K),K=1,15),THL,ZHL
25 FORMAT(11,F4.0,F5.0,1X,15A1,1X,F6.0,F4.0)
WRITE(3,26) IBASE,X(II),Y(II),(NAVN(K),K=1,15),THL,ZHL
26 FORMAT(1X,11,F4.0,F5.0,1X,15A1,1X,F6.0,F4.0)
X(II)=X(II)/1000.
Y(II)=(Y(II)-6000.)/1000.
DO 22 I=1,13
READ(8,23,END=24) I,(CCI(K,I,II),K=1,4)
22 CONTINUE
IKURV1=IKURV
CALL INVERS(SARR,10,IKURV,SINV,ZVEKT,IVEKT,IERR)
C BEREGNING AV STEDMATRISE FOR PUNKTET X,Y OG 3 NABOPUNKTER-----
40 CALL LFEED(2)
C
23 FORMAT(1X,I2,1X,8E15.9)
22 CONTINUE
24 ENDFILE 8
CALL LFEED(1)
21 CONTINUE
C INNLESNING AV HOVEDFLATE-----
HFILE(5)=48+IBASE
CALL OPEN(7,HFILE,0)
READ(7,15) IHORD
15 FORMAT(I15)
IH=IHORD+1
DO 16 I=1,IH
READ(7,17) H(I,J),J=1,IH)
17 FORMAT(6F15.9)
16 CONTINUE
ENDFILE 7
C INVERTERING AV STEDMATRISE-----
30 DO 31 I=1,10
DO 31 J=1,10
SINV(I,J)=0.0
31 CONTINUE
IF( IKURV-2) 32,33,34
C 1 KURVE-----
32 SINV(1,1)=1.0
IKURV1=1
GOTO 40
C 2 KURVER-----
33 FKONST=(X(1)-X(2))**2+(Y(1)-Y(2))**2
SINV(1,1)=(X(2)-X(1))*X(2)+(Y(2)-Y(1))*Y(2)/FKONST
SINV(1,2)=(X(1)-X(2))*X(1)+(Y(1)-Y(2))*Y(1)/FKONST
SINV(1,3)=0.0
SINV(2,1)=(X(1)-X(2))/FKONST
SINV(2,2)=(X(2)-X(1))/FKONST
SINV(2,3)=0.0
SINV(3,1)=(Y(1)-Y(2))/FKONST
SINV(3,2)=(Y(2)-Y(1))/FKONST
SINV(3,3)=0.0
IKURV1=3
GOTO 40
C FLERE KURVER-----
34 DO 35 I=1,IKURV
SARR(I,1)=1.
SARR(I,2)=X(I)
SARR(I,3)=Y(I)
SARR(I,4)=X(I)*X(I)
SARR(I,5)=X(I)*Y(I)
SARR(I,6)=Y(I)*Y(I)
SARR(I,7)=X(I)*X(I)*X(I)
SARR(I,8)=X(I)*X(I)*Y(I)
SARR(I,9)=X(I)*Y(I)*Y(I)
SARR(I,10)=Y(I)*Y(I)*Y(I)
35 CONTINUE
IKURV1=IKURV
CALL INVERS(SARR,10,IKURV,SINV,ZVEKT,IVEKT,IERR)
C BEREGNING AV STEDMATRISE FOR PUNKTET X,Y OG 3 NABOPUNKTER-----
40 CALL LFEED(2)

```

```

IKOORD=IREAD(' Gi X-koordinaten til stedet: ',30)
IF(IKOORD.LE.0) GOTO 10
JKOORD=IREAD(' Gi Y-koordinaten til stedet: ',30)
XKOORD=IKOORD
YKOORD=JKOORD
XKOORD=XKOORD/1000.
YKOORD=YKOORD-6000.)/1000.
DO 41 II=1,2
DO 41 JJ=1,2
  KK=2*(JJ-1)+II
  X(II)=XKOORD+(II-1)*0.001
  Y(JJ)=YKOORD+(JJ-1)*0.001
  ZHL=ZHLCAL(X(II),Y(JJ),IHORD,H)
  SVEKT(1)=1.
  SVEKT(2)=X(II)
  SVEKT(3)=Y(II)
  SVEKT(4)=X(II)*X(II)
  SVEKT(5)=X(II)*Y(II)
  SVEKT(6)=Y(II)*Y(II)
  SVEKT(7)=X(II)*X(II)*X(II)
  SVEKT(8)=X(II)*X(II)*Y(II)
  SVEKT(9)=X(II)*Y(II)*Y(II)
  SVEKT(10)=Y(II)*Y(II)*Y(II)
  DO 43 J=1,IKURV
    AAASUM=0.0
    DO 44 K=1,IKURV1
      CCC=SINV(K,J)
      DDD=SVEKT(K)
      AAASUM=AAASUM+CCC*DDD
    44 CONTINUE
    ZVEKT(J)=AAASUM
  43 CONTINUE
C BEREGNING AV KOEFFISIENTMATRISE FOR STEDET X, Y-----
DO 45 K=1,13
DO 45 I=1,IORD
  AAASUM=0.0
  DO 46 J=1,IKURV
    BBB=CC1(I,K,J)
    CCC=ZVEKT(J)
    AAASUM=AAASUM+BBB*CCC
  46 CONTINUE
  C1(I,K,KK)=AAASUM*ZHL
  45 CONTINUE
  41 CONTINUE
C AUTUMATISK GRADIENTBEREGNING-----
DO 47 III=8,26
  TID=500*III
  ITID=TID/1000.
  DT=TID/1000.-ITID
  DO 48 II=1,2
    DO 48 JJ=1,2
      KK=2*(JJ-1)+II
      Z(II,JJ)=C1(1,ITID,KK)+C1(2,ITID,KK)*DT
      +C1(3,ITID,KK)*DT+C1(4,ITID,KK)*DT*DT*DT
    48 CONTINUE
    DX=Z(2,1)-Z(1,1)
    DY=Z(1,2)-Z(1,1)
    GRAD=SQRT(DX*DX+DY*DY)
    TETA=ATAN2(DX,DY)*(180/3.141592)
    WRITE(3,49) TID,Z(1,1),GRAD,TETA
  49 FORMAT('Tid: ',F6.0,' Høyde: ',F6.1,' Gradient: ',
    * ,F6.2,' Vinkel:',F5.1)
  47 CONTINUE
C GRADIENTBEREGNING-----
50 CALL LFEED(2)
  TID=IREAD(' Gi tidspunktet: ',18)
  IF(TID.LE.0.0) GOTO 40
  ITID=TID/1000.
  DT=TID/1000.-ITID
  DO 51 II=1,2
  DO 51 JJ=1,2
    KK=2*(JJ-1)+II
    * Z(II,JJ)=C1(1,ITID,KK)+C1(2,ITID,KK)*DT
    +C1(3,ITID,KK)*DT+C1(4,ITID,KK)*DT*DT*DT
  51 CONTINUE
  DX=Z(2,1)-Z(1,1)
  DY=Z(1,2)-Z(1,1)
  GRAD=SQRT(DX*DX+DY*DY)
  TETA=ATAN2(DX,DY)*(180/3.141592)
  WRITE(3,52) Z(1,1),GRAD,TETA
  52 FORMAT(' Høyde: ',F6.1,' Gradient: ',F6.2,' Vinkel: ',F5.1)
  GOTO 50
END

```



```

C*****
C Program for datering av MG-lokaliteter i spline-modell
C
C Programmet av Kjell Kjenstad, Uio
C Juni 1984
C
C*****
PROGRAM MGCALC
  BYTE HFILE(11),MFILE(11),FILE(11),NAVN(15),DANAVN(15)
  *,FORF(3)
  REAL SARR(10,10),X(10),Y(10),SINV(10,10),ZVEKT(10),SVEKT(10)
  *,DVEKT(7),H(5,5),CC1(4,13,10),CC2(4,13,10),C1(4,13),C2(4,13)
  DOUBLE PRECISION AAASUM,BBSUM,AAAHOF,AAADIF,BBB,CCC,DDD
  INTEGER IVEKT(10)
  DATA HFILE/'H','L',2*'X',4*' ','M','A','T'/'
  DATA MFILE/'M','G',2*'X',4*' ','D','A','T'/'
C-----
10 CALL TDV(25)
  WRITE(3,11)
11 FORMAT(1X, /
  *,8X, '*****'/'
  *,8X, '***'/'
  *,8X, '**Program for datering av MG-lokaliteter i modell basert**/'
  *,8X, '*** på flere spline-tilpassede strandforskningskurver **/'
  *,8X, '***'/'
  *,8X, '*****'/'
C NULLSTILLING-----
DO 13 I=1,4
  DO 13 J=1,13
    C1(I,J)=0.0
    C2(I,J)=0.0
    DO 13 K=1,10
      CC1(I,J,K)=0.0
      CC2(I,J,K)=0.0
13 CONTINUE
C INNLESNING AV SPLINE-PARAMETERFILER-----
20 IORD=4
IKURV=IREAD(' Gi antallet kurver som skal med i modellen: ',45)
IF((IKURV.LE.0).OR.(IKURV.GT.10)) CALL FCHAIN('MENYPRXXCOM',0)
CALL TWRITE(' Gi navnet på datafilene: ',30)
CALL LFEED(1)
DO 21 II=1,IKURV
  CALL FREAD(FILE,'INNKURVE.DAT')
  CALL LFEED(1)
  IF(FILE(1).EQ.32) GOTO 10
  FILE(9)='S'
  FILE(11)='L'
  FILE(10)='P'
  CALL OPEN(8,FILE,0)
  READ(8,25,FRR=10) IBASE,X(II),Y(II),(NAVN(K),K=1,15),THL,ZHL
25 FORMAT(11,F4.0,F5.0,1X,15A1,1X,F6.0,F4.0)
  WRITE(3,26) IBASE,X(II),Y(II),(NAVN(K),K=1,15),THL,ZHL
26 FORMAT(1X,11,F4.0,F5.0,1X,15A1,1X,F6.0,F4.0)
  X(II)=X(II)/1000.
  Y(II)=(Y(II)-6000.)/1000.
C FLERE KURVER-----
34 DO 35 I=1,IKURV
  SARR(I,1)=1.
  SARR(I,2)=X(I)
  SARR(I,3)=Y(I)
  SARR(I,4)=X(I)*X(I)
  SARR(I,5)=X(I)*Y(I)
  SARR(I,6)=Y(I)*Y(I)
  SARR(I,7)=X(I)*X(I)*X(I)
  SARR(I,8)=X(I)*X(I)*Y(I)
  SARR(I,9)=X(I)*Y(I)*Y(I)
  SARR(I,10)=Y(I)*Y(I)*Y(I)
35 CONTINUE
  IKURV1=IKURV
  CALL INVERS(SARR,10,IKURV,SINV,ZVEKT,IVEKT,IERR)
DO 22 I=1,13
  READ(8,23,END=24) I,(CC1(K,I,II),K=1,4),(CC2(K,I,II),K=1,4)
23 FORMAT(1X,I2,1X,8E15.9)
22 CONTINUE
24 ENDFILE 8
  CALL LFEED(1)
21 CONTINUE
C INNLESNING AV HOVEDPLATE-----
HFILE(5)=48+IBASE
CALL OPEN(7,HFILE,0)
READ(7,15) IHORD
15 FORMAT(I15)
IH=IHORD+1
DO 16 I=1,IH
  READ(7,17) (H(I,J),J=1,IH)
17 FORMAT(6F15.9)
16 CONTINUE
  ENDFILE 7
C INVERTERING AV STEDMATRISE-----
30 DO 31 I=1,10
  DO 31 J=1,10
    SINV(I,J)=0.0
31 CONTINUE
  IF( (IKURV-2) 32,33,34)
C 1 KURVE-----
32 SINV(1,1)=1.0
  IKURV1=1
  GOTO 37
C 2 KURVER-----
33 FKONST=(X(1)-X(2))**2+(Y(1)-Y(2))**2
  SINV(1,1)=(X(2)-X(1))*X(2)+(Y(2)-Y(1))*Y(2)/FKONST
  SINV(1,2)=((X(1)-X(2))*X(1)+(Y(1)-Y(2))*Y(1))/FKONST
  SINV(1,3)=0.
  SINV(2,1)=(X(1)-X(2))/FKONST
  SINV(2,2)=(X(2)-X(1))/FKONST
  SINV(2,3)=0.
  SINV(3,1)=(Y(1)-Y(2))/FKONST
  SINV(3,2)=(Y(2)-Y(1))/FKONST
  SINV(3,3)=0.
  IKURV1=3
  GOTO 37
C FLERE KURVER-----
34 DO 35 I=1,IKURV
  SARR(I,1)=1.
  SARR(I,2)=X(I)
  SARR(I,3)=Y(I)
  SARR(I,4)=X(I)*X(I)
  SARR(I,5)=X(I)*Y(I)
  SARR(I,6)=Y(I)*Y(I)
  SARR(I,7)=X(I)*X(I)*X(I)
  SARR(I,8)=X(I)*X(I)*Y(I)
  SARR(I,9)=X(I)*Y(I)*Y(I)
  SARR(I,10)=Y(I)*Y(I)*Y(I)
35 CONTINUE
  IKURV1=IKURV
  CALL INVERS(SARR,10,IKURV,SINV,ZVEKT,IVEKT,IERR)

```

```

C ÅPNING AV MG-LOKALITETS-FIL-----
C 37 MFILE(5)=48+IBASE
  37 MFILE(5)=48+6
  CALL LFEED(2)
  WRITE(3,38) (MFILE(K),K=1,11)
  38 FORMAT(' Inputfil: ',8A1,' ',3A1)
  CALL LFEED(1)
  CALL OPEN(7,MFILE,0)
  MFILE(3)='Y'
  MFILE(4)='Y'
  WRITE(3,39) (MFILE(K),K=1,11)
  39 FORMAT(' Outputfil: ',8A1,' ',3A1)
  CALL OPEN(8,MFILE,0)
  CALL LFEED(2)
C PARAMETERDIALOG-----
  DZHL=IREAD(' Gi standardavvik på hovedflata: ',32)
  IGRAD=IREAD(' Gi usikkerhetsgradient: ',25)
  IANT=IREAD(' Gi minste nummerverdi: ',24)
C LESELØKKE-----
  40 IANT= IANT+1
  READ(7,41,ERR=90,END=90) XX,YY,(NAVN(K),K=1,15),ZZ
  *,(FORF(K),K=1,3)
  41 FORMAT(2F5.0,1X,15A1,1X,F4.0,5X,3A1)
  HOYD=ZZ
  XKOORD=(XX-1000.*IBASE)/1000.
  YKOORD=(YY-6000.)/1000.
  SVEKT(1)=1.
  SVEKT(2)=XKOORD
  SVEKT(3)=YKOORD
  SVEKT(4)=XKOORD*XKOORD
  SVEKT(5)=XKOORD*YKOORD
  SVEKT(6)=YKOORD*YKOORD
  SVEKT(7)=XKOORD*XKOORD*XKOORD
  SVEKT(8)=XKOORD*XKOORD*YKOORD
  SVEKT(9)=YKOORD*YKOORD*YKOORD
  SVEKT(10)=YKOORD*YKOORD*XKOORD
  DO 343 J=1, IKURV
    AAASUM=0.0
    DO 344 K=1, IKURV1
      CCC=SINV(K,J)
      DDD=SVEKT(K)
      AAASUM=AAASUM+CCC*DDD
    344 CONTINUE
      ZVEKT(J)=AAASUM
    343 CONTINUE
  C BEREGNING AV HOVEDFLATA FOR STEDET X Y-----
  ZZHL=ZHL*CAL(XKOORD,YKOORD,IHOR,H)
  C BEREGNING AV KOEFFISIENTMATRISE FOR STEDET X,Y-----
  DO 351 K=1,13
  DO 351 I=1,IORD
    AAASUM=0.0
    DO 352 J=1,IKURV
      BBB=CC1(I,K,J)
      CCC=ZVEKT(J)
      AAASUM=AAASUM+BBB*CCC
    352 CONTINUE

```

```

  C1(I,K)=AAASUM*ZZHL
  351 CONTINUE
  C BEREGNING AV COVARIANSMATRISE FOR STEDET X,Y-----
  IF(IKURV-2) 360,370,380
  C EN KURVE-----
  360 ZVEKT(1)=1+IGRAD*(SQRT((XKOORD-X(1))^2+(YKOORD-Y(1))^2))
  GOTO 380
  C TO KURVER-----
  370 DIST=(2*XKOORD-X(1)-X(2))^2+(2*YKOORD-Y(1)-Y(2))^2
  DIST=SQRT(DIST/FKONST)
  ZVEKT(1)=0.5*(1-DIST)
  ZVEKT(2)=0.5*(1+DIST)
  C FLERE KURVER-----
  380 DO 381 K=1,13
  DO 381 I=1,IORD
    AAASUM=0.0
    DO 382 J=1,IKURV
      BBB=CC2(I,K,J)
      CCC=ZVEKT(J)
      AAASUM=AAASUM+BBB*CCC*CCC
    382 CONTINUE
      C2(I,K)=AAASUM*ZZHL
    381 CONTINUE
  C TIDSPUNKTBBEREGNING-----
  50 USIKK=1.
  IF(FORF(3).EQ.' ') USIKK=3.
  IF(FORF(2).EQ.' ') USIKK=5.
  IF(FORF(1).NE.'*') USIKK=3.
  TIDMIN=8000.
  TID0=TCAL(HOYD,TIDMIN,C1)
  DHOYD=HCAL(TID0,C2)
  TID1=TCAL((HOYD-DHOYD)-DZHL-USIKK),(TIDMIN-1000),C1)
  TID2=TCAL((HOYD+DHOYD)+DZHL+USIKK),TIDMIN,C1)
  TID1=ABS(TID0-TID1)
  TID2=ABS(TID2-TID0)
  TIDM=(TID1+TID2)/2.
  IT=TID0/10
  TID0=10*IT
  IT=TIDM/10
  TIDM=10*IT
  IF(TIDM.GE.1000.) TIDM=999
  IF(TIDM.LE.-100.) TIDM=999
  WRITE(8,52) IANT,(NAVN(K),K=1,15),XX,YY,ZZ,ZZHL,TID0,TIDM
  *,(FORF(K),K=1,3)
  52 FORMAT(1X,I3,1X,15A1,2F6.0,2F5.0,F7.0,'+- ',F4.0,1X,3A1)
  WRITE(3,52) IANT,(NAVN(K),K=1,15),XX,YY,ZZ,ZZHL,TID0,TIDM
  *,(FORF(K),K=1,3)
  GOTO 40
  90 ENDFILE 7
  ENDFILE 8
  CALL FCHAIN('MENYPRXXCOM',0)
  END
C*****
C BEREGNING AV TIDSPUNKT I HAVMODELL
C
C

```

```
C*****
FUNCTION TCAL(HOYD,TIDMIN,C1)
REAL C1(4,13)
IMIN=TIDMIN
DO 21 I=IMIN,14000,1000
  TID1=I-1000
  TID2=I
  HOYD1=HCAL(TID1,C1)
  HOYD2=HCAL(TID2,C1)
  IF(((HOYD.GE.HOYD2).AND.(HOYD.GE.HOYD1))
    * .OR.((HOYD.LE.HOYD2).AND.(HOYD.LE.HOYD1))) GOTO 21
  DO 22 J=1,7
    TIDM=(TID1+TID2)/2
    HOYD1=HCAL(TID1,C1)
    HOYD2=HCAL(TID2,C1)
    HOYDM=HCAL(TIDM,C1)
    IF(((HOYD.LE.HOYD2).AND.(HOYD.GT.HOYDM))
      * .OR.((HOYD.GE.HOYD2).AND.(HOYD.LT.HOYDM))) GOTO 23
    IF(((HOYD.LE.HOYD1).AND.(HOYD.GT.HOYDM))
      * .OR.((HOYD.GE.HOYD1).AND.(HOYD.LT.HOYDM))) GOTO 24
  23   TID1=TIDM
      GOTO 22
  24   TID2=TIDM
      GOTO 22
  22 CONTINUE
  TCAL=TIDM
  RETURN
21 CONTINUE
TCAL=14000
RETURN
END
C*****
C
C HØYDEBEREGNINGSFUNKJON
C *****
FUNCTION HCAL(TID,C1)
REAL C1(4,13)
ITID=TID/1000.
DT=TID/1000.-ITID
HCAL=
*C1(1,ITID)+C1(2,ITID)*DT+C1(3,ITID)*DT*DT+C1(4,ITID)*DT*DT*DT
RETURN
END
```

```

*****
**
** PROGRAMMER SOM FORETAR ESTIMERING OG
** BEREGNINGER ETTER DEN ENKLE HAVNIVÅMODELLEN
**
*****
MENYPRY: er meny-programmet for den
enkle havnivå-pakken.....s. A - 20

KURVB EY: foretar regressjonstilpasning
etter den enkle modellen av
strandforykningskurver.....s. A - 21

KURVPL Y: foretar plottning av strand-
forykningskurver etter den
enkle modellen på grafisk skjerm.....s. A - 22

MODEBEY: foretar beregning av glatt overgang
mellom strandforykningskurver
beregnet etter den enkle modellen.....s. A - 24

FLATBEY: foretar direkte beregning av
parametriserte trendflatehøyder.....s. A - 28

GRADIEY: foretar beregning av flategradienter og
flateretninger på et sted
etter den enkle havnivåmodellen.....s. A - 28

MGCALCY: foretar datering av MG-lokaliteter på
grunnlag av den enkle havnivåmodellen.....s. A - 30

```

```

C*****
C
C Menyprogram for HAVNIVÅMODEL- programmer
C
C Programmert av Kjell Kjenstad, UiO
C Januar 1983
C
C*****
C PROGRAM MODELL
C-----
11 CALL TDV(25)
WRITE(3,12)
12 FORMAT(1X, / /
*,8X, '**'
*,8X, '** HOVEDMENY
*,8X, '**
*,8X, '** Programmer for beregning og
*,8X, '** plottning av havnivåmodeller
*,8X, '**
*,8X, '**
*,8X, '**
*,8X, '**
*,8X, '** 0: Slutt
*,8X, '**
*,8X, '** 1: Kurvetilpasning
*,8X, '** 2: Kurveplottning av temporærfil
*,8X, '** 3: Rom-modell-beregning
*,8X, '** 4: Beregning av flatehøyder
*,8X, '** 5: Beregning av gradienter og retninger
*,8X, '** 6: Datering av MG-lokaliteter
*,8X, '** 7: Hopp over til kartplotteprogrammer
*,8X, '**
*,8X, '** :')
//
READ(3,13) NEXT
13 FORMAT(I2)
CALL TDV(25)
IF(NEXT.LE.0) STOP
IF(NEXT.GT.6) GOTO 11
GOTO (10,20,30,40,50,60),NEXT
C-----
10 CALL FCHAIN('KURVB EYCOM',0)
20 CALL FCHAIN('KURVPLYCOM',0)
30 CALL FCHAIN('MODEBEYCOM',0)
40 CALL FCHAIN('FLATBEYCOM',0)
50 CALL FCHAIN('GRADIEYCOM',0)
60 CALL FCHAIN('MGCALCYCOM',0)
60 CALL FCHAIN('MENYPRZCOM',0)
C-----
END

```



```

36 WRITE(3,36) DMAT(I), (COVARR(I,K),K=1,IORD)
37 FORMAT(1X,8E10.2)
38 CONTINUE
39 ITEMP=AREAD(' <CR>',5)
40 CALL TDV(25)
41 IF((NEXT.EQ.2).OR.(NEXT.EQ.5)) GOTO 50
42 IF((NEXT.EQ.1).OR.(NEXT.EQ.4)) GOTO 60
C UTSKRIFT-----
43 FILE(9)='M'
44 CALL OPEN(7,FILE,0)
45 WRITE(7,25) X,Y,(NAVN(K),K=1,15),THL,ZHL
46 DO 41 I=1,7
47   WRITE(7,42) DMAT(I),(COVARR(I,K),K=1,7)
48   FORMAT(1X,8E15.9)
49 CONTINUE
50 ENDFILE 7
51 ITEMP=AREAD(' <CR>',5)
52 CALL TDV(25)
53 GOTO 10
C PLOTTING-----
54 CALL OPEN(8,'KURVMAT TMP',0)
55 WRITE(8,53) IORD,ILOG
56 FORMAT(1X,2I1)
57 WRITE(8,25) X,Y,(NAVN(K),K=1,15),THL,ZHL
58 DO 51 I=1,7
59   WRITE(8,52) DMAT(I),(COVARR(I,K),K=1,7)
60   FORMAT(1X,8E15.9)
61 CONTINUE
62 ENDFILE 8
63 CALL TDV(25)
64 CALL FCHAIN('KURVPLYCOM',0)
65 HØYDEBEREGNING-----
66 ITID=IREAD(' Gi tidspunkt: ',15)
67 IF(ITID.EQ.0) GOTO 10
68 TID=ITID
69 ZNO=HOYCAL(TID,IORD,DMAT)
70 ZNOVAR=VARCAL(TID,IORD,COVARR)
71 IF(ILOG.EQ.0) CALL LINTRA(HOYD,HOYD1,HOYD2,ZHL,ZHLVAR,ZNO,ZNOVAR)
72 IF(ILOG.EQ.1) CALL LOGTRA(HOYD,HOYD1,HOYD2,ZHL,ZHLVAR,ZNO,ZNOVAR)
73 WRITE(3,61) HOYD
74 FORMAT('/',Høyden er: ',F5.1//)
75 GOTO 60
C HOPP TIL MODELPROGRAM-----
76 CALL FCHAIN('MODEBRYCOM',0)
C SLUTT-----
END

C *****
C PROGRAM FOR PLOTTING AV KURVER
C Programmert av Kjell Kjenstad, Uio
C Januar 1983
C *****
PROGRAM KURVPL
  BYTE FILE(11),NAVN(15),DANAVN(15)
  REAL T(20),Z(20),DMAT(7),COVARR(7,7)
C LESING PÅ DATAFIL-----
  CALL OPEN(6,'KURV DAT TMP',0)
  READ(6,23) X,Y,(NAVN(K),K=1,15),THL,ZHL
  DO 22 I=1,20
  READ(6,23,END=24) XX,YY,(DANAVN(K),K=1,15),T(I),Z(I)
  FORMAT(2F5.0,1X,15A1,1X,F6.0,F4.0)
  IF(DANAVN(1).EQ.102) I=I-1
  CONTINUE
  IDAT=I-1
  ENDFILE 6
C LESING AV MATRISEFIL-----
  CALL OPEN(7,'KURVMAT TMP',0)
  READ(7,44) IORD,ILOG
  FORMAT(2I1)
  READ(7,23) X,Y,(NAVN(K),K=1,15),THL,ZHL
  DO 41 I=1,IORD
  READ(7,42) DMAT(I),(COVARR(I,K),K=1,IORD)
  FORMAT(8E15.9)
  CONTINUE
  ENDFILE 7
C DIALOG-----
  CALL TDV(25)
  WRITE(3,56)
  FORMAT(
    *.8X,*****
    *.8X,***          Program for plotting av
    *.8X,***          strandforskyvningskurver
    *.8X,***
    *.8X,*****
    *//)
  NEXT=IREAD(' Standard-diagram (Ja)',22)
  JMG=IREAD(' Gi MG-høyden: ',15)
  ISD=IREAD(' Gi Hovedflatens standardavvik: ',32)
  ZHLVAR=ISD*ISD
  IF(JMG.LE.0) JMG=240
  IMIN=0
  IMAX=13000
  IVAR=1
  CALL GRAFIX
  IF(NEXT) 70,51,52
  CALL INIT
  CALL RES(1)
  CALL DEFCLR(0,0,10)
  CALL RAMME(JMG,JMAX,NAVN)
  END

```

```

GOTO 60
51 NEXT=IREAD(' Utvisking (Ja)',15)
IF(NEXT) 50,53,54
53 CALL WORKON(0)
CALL RES(1)
CALL UNSCAL
CALL DEFCLR(0,0,0,10)
CALL RAMME(JMG,JMAX,NAVN)
GOTO 55

54 CALL INIT
CALL RES(1)
CALL DEFCLR(0,0,0,10)
CALL RAMME(JMG,JMAX,NAVN)
IVAR=IREAD(' SD-grenser (Ja)',16)
IMIN=IREAD(' Gi nedre tidsgrense: ',22)
IMAX=IREAD(' Gi øvre tidsgrense: ',22)
IF(IMAX.LE.IMIN) IMAX=12900
C PLOTTING AV DATERTE NIVÅ-----
60 IF(JMG.GT.JMAX) JMG=JMAX
IHFAKT=240/JMAX
IDATMI=IDAT-1
DO 62 I=1, IDATMI
II0=(J3000-T(I))/25+100
JJ0=IHFAKT*Z(I)+140
II1=II0-4
II2=II0+4
JJ1=JJ0-2
JJ2=JJ0+3
CALL HXLINE(II1,JJ0,II2,JJ0,1)
CALL HXLINE(II0,JJ1,II0,JJ2,1)
62 CONTINUE

C PLOTTING AV KURVE-----
DO 61 I=1,259
TID=13000-50*I
II0=2*I+100
IF(TID.LT.IMIN) GOTO 70
ZNO=HOYCAL(TID,IORD,DMAT)
ZNOVAR=VARCAL(TID,IORD,COVARR)
IF(ILOG.EQ.0)
* CALL LINTRA(HOYD,HOYD1,HOYD2,ZHL,ZHLVAR,ZNO,ZNOVAR)
IF(ILOG.EQ.1)
* CALL LOGTRA(HOYD,HOYD1,HOYD2,ZHL,ZHLVAR,ZNO,ZNOVAR)
JJ0=IHFAKT*HOYD+140
JJ1=IHFAKT*HOYD1+140
JJ2=IHFAKT*HOYD2+140
IF(TID.GT.IMAX) GOTO 65
IF(HOYD.LT.JMG) CALL HXLINE(IGL0,JGL0,II0,JJ0,1)
IF(IVAR.EQ.0) GOTO 65
IF(HOYD.LT.JMG) CALL HXLINE(IGL0,JGL1,II0,JJ1,1)
IF(HOYD.LT.JMG) CALL HXLINE(IGL0,JGL2,II0,JJ2,1)
C-----OPPDATERING-----
65 IGL0=II0
JGL0=JJ0
JGL1=JJ1
JGL2=JJ2

JGL2=JJ2
61 CONTINUE
C NESTE DIAGRAM-----
70 CALL TDV(25)
WRITE(3,71)
71 FORMAT(1X,///
*.8X,'*****')
*.8X,'**'
*.8X,'**'
*.8X,'**'
*.8X,'**'
*.8X,'**'
*.8X,'**'
*.8X,'**'
*///)
NEXT=IREAD(' ',10)
CALL TDV(25)
IF(NEXT-1) 73,50,74
73 IF(IDAT.GT.0) CALL FCHAIN('KURVBEYYCOM',0)
IF(IDAT.EQ.0) CALL FCHAIN('MODEBEYYCOM',0)
74 CALL FCHAIN('MENYPRYCOM',0)
C SLUTT-----
END
C SUBROUTINE DIAGRAMME*****
SUBROUTINE RAMME(JMG,JMAX,NAVN)
BYTE NAVN(15),TONAVN(46)
DATA TONAVN/'S','H','O','R','E',' ','D','I','S','P','L','A','
'C','E','M','E','N','T',' ','C','U','R','V','E',' ','F','R','
'O','M',' ',' ',' ',' ',' ',' ',' ',' ',' ',' ',' ',' ',' ','
',' ',' ','
TONAVN(46)='^'
DO 21 I=31,45
TONAVN(I)=NAVN(I-30)
21 CONTINUE
CALL HXTEXT(200,440,1,TONAVN)
CALL HXTEXT(87,70,1,13^1)
CALL HXTEXT(127,70,1,12^1)
CALL HXTEXT(167,70,1,11^1)
CALL HXTEXT(207,70,1,10^1)
CALL HXTEXT(257,70,1,9^1)
CALL HXTEXT(297,70,1,8^1)
CALL HXTEXT(337,70,1,7^1)
CALL HXTEXT(377,70,1,6^1)
CALL HXTEXT(417,70,1,5^1)
CALL HXTEXT(457,70,1,4^1)
CALL HXTEXT(497,70,1,3^1)
CALL HXTEXT(537,70,1,2^1)
CALL HXTEXT(577,70,1,1^1)
CALL HXTEXT(617,70,1,0^1)
CALL HXTEXT(500,40,1,1000 YEARS BP^1)
DO 31 I=0,13
II=2*I*20+100
CALL HXLINE(II,94,II,380,1)
31 CONTINUE

```



```
CALL LFEED(1)
DO 22 I=1,IORD
  READ(6,23) DMAT(I,II), (COVARR(I,K,II),K=1,7)
  WRITE(3,23) DMAT(I,II), (COVARR(I,K,II),K=1,IORD)
  FORMAT(8E15.9)
23 CONTINUE
22 ENDFILE 6
  CALL LFEED(2)
  ENDFILE 9
  CALL LFEED(30)
  C INVERTERING AV STEDMATRISE-----
  DO 30 I=1,I0
  DO 31 J=1,I0
    SINV(I,J)=0.0
  31 CONTINUE
  IF(IKURV-2) 310,320,330
  C 1 KURVE-----
  310 SINV(1,1)=1.0
  IKURV1=1
  GOTO 340
  C 2 KURVER-----
  320 FKONST=(X(1)-X(2))*2+(Y(1)-Y(2))*2
  SINV(1,1)=((X(2)-X(1))*X(2)+(Y(2)-Y(1))*Y(2))/FKONST
  SINV(1,2)=((X(1)-X(2))*X(1)+(Y(1)-Y(2))*Y(1))/FKONST
  SINV(1,3)=0.
  SINV(2,1)=(X(1)-X(2))/FKONST
  SINV(2,2)=(X(2)-X(1))/FKONST
  SINV(2,3)=0.
  SINV(3,1)=(Y(1)-Y(2))/FKONST
  SINV(3,2)=(Y(2)-Y(1))/FKONST
  SINV(3,3)=0.
  IKURV1=3
  GOTO 340
  C FLERE KURVER-----
  330 DO 331 I=1,IKURV
  SARR(I,1)=1.
  SARR(I,2)=X(I)
  SARR(I,3)=Y(I)
  SARR(I,4)=X(I)*X(I)
  SARR(I,5)=X(I)*Y(I)
  SARR(I,6)=Y(I)*Y(I)
  SARR(I,7)=X(I)*X(I)*X(I)
  SARR(I,8)=X(I)*X(I)*Y(I)
  SARR(I,9)=X(I)*Y(I)*Y(I)
  SARR(I,10)=Y(I)*Y(I)*Y(I)
  331 CONTINUE
  IKURV1=IKURV
  CALL INVERS(SARR,10,IKURV,SINV,ZVEKT,IVEKT,IEFRR)
  CALL LFEED(5)
  IF((NEXT.EQ.4).OR.(NEXT.EQ.8)) GOTO 70
  C BEREGNING AV STEDMATRISE FOR PUNKTET X,Y-----
  340 ILOG=IREAD(' Logaritmetransformasjon (Ja)',29)
  IKOORD=IREAD(' Gi X-koordinaten til stedet: ',30)
  IF(IKOORD.LE.0) GOTO 10
  JKOORD=IREAD(' Gi Y-koordinaten til stedet: ',30)
```

```

380 DO 381 I=1,IORD
DO 381 J=1,IORD
AAASUM=0.0
DO 382 K=1,IKURV
BBB=COVARR(I,J,K)
CCC=ZVEKT(K)
AAASUM=AAASUM+BBB*CCC*CCC
382 CONTINUE
COV(I,J)=AAASUM
381 CONTINUE
CALL LFEED(2)
ISD=IREAD(' Gi standardavvik på hovedflata: ',32)
ZHLVAR=ISD*ISD
CALL OPEN(7,HFILE,0)
READ(7,391) IHORD
391 FORMAT(I15)
IH=IHORD+1
DO 392 I=1,IH
READ(7,393) (H(I,J),J=1,IH)
393 FORMAT(6F15.9)
392 CONTINUE
ENDFILLE 7
ZHL=ZHLCAL(XKOORD,YKOORD,IHORD,H)
WRITE(3,394) ZHL
394 FORMAT(/' Hovedlinjehøyden er: ',F7.2////////)
C VALG-----
C HØYDEBEREGNINGSLØKKE-----
GOTO (40,50,60,70,40,50,60,70),NEXT
40 ITID=IREAD(' Gi tidspunkt: ',15)
IF(ITID.LE.0) GOTO 10
TID=ITID
ZNO=HOYCAL(TID,IORD,DMAT)
ZNOVAR=VARCAL(TID,IORD,COV)
IF(ILOG.EQ.0) CALL LINTRA(HOYD,HOYD1,HOYD2,ZHL,ZHLVAR,ZNO,ZNOVAR)
IF(ILOG.EQ.1) CALL LOGTRA(HOYD,HOYD1,HOYD2,ZHL,ZHLVAR,ZNO,ZNOVAR)
HOYD1=HOYD-HOYD1
WRITE(3,42) HOYD,HOYD1
42 FORMAT(' Høyden er: ',F5.1,' +- ',F4.1,' m o.h.////////)
GOTO 40
C TIDSPUNKTSBEREGNING-----
50 IHOYD=IREAD(' Gi høyden på stedet: ',22)
IHOYD=IHOYD
IF(IHOYD.EQ.0) GOTO 10
TIDMIN=1000.
IF(ILOG.EQ.0) ZNO=HOYD/ZHL
IF(ILOG.EQ.1) ZNO=ALOG(HOYD+1)/ALOG(ZHL+1)
TID0=TIDCAL(ZNO,TIDMIN,IORD,DVEKT)
IF(TID0.GE.15000.) GOTO 50
TID=TID0
ZNOVAR=VARCAL(TID,IORD,COV)
IF(ILOG.EQ.0) CALL LINTRA(HOYD,HOYD1,HOYD2,ZHL,ZHLVAR,ZNO,ZNOVAR)
IF(ILOG.EQ.1) CALL LOGTRA(HOYD,HOYD1,HOYD2,ZHL,ZHLVAR,ZNO,ZNOVAR)
IF(ILOG.EQ.0) ZNO1=HOYD1/ZHL
IF(ILOG.EQ.1) ZNO1=ALOG(HOYD1+1)/ALOG(ZHL+1)
IF(ILOG.EQ.0) ZNO2=HOYD2/ZHL

```

```

XKOORD=XKOORD
YKOORD=YKOORD
XKOORD=XKOORD/1000.
YKOORD=(YKOORD-6000.)/1000.
DO 341 I=1,10
SVEKT(I)=0.0
341 CONTINUE
DO 342 I=1,7
DVEKT(I)=0.0
DO 342 J=1,7
COV(I,J)=0.0
342 CONTINUE
SVEKT(1)=1.
SVEKT(2)=XKOORD
SVEKT(3)=YKOORD
SVEKT(4)=XKOORD*XKOORD
SVEKT(5)=XKOORD*YKOORD
SVEKT(6)=YKOORD*YKOORD
SVEKT(7)=XKOORD*XKOORD*XKOORD
SVEKT(8)=XKOORD*XKOORD*YKOORD
SVEKT(9)=XKOORD*YKOORD*YKOORD
SVEKT(10)=YKOORD*YKOORD*YKOORD
DO 343 J=1,IKURV
AAASUM=0.0
DO 344 K=1,IKURV1
CCC=SINV(K,J)
DDD=SVEKT(K)
AAASUM=AAASUM+CCC*DDD
344 CONTINUE
ZVEKT(J)=AAASUM
WRITE(3,345) J,ZVEKT(J)
345 FORMAT(1X,I1,'. faktor: ',F8.4)
343 CONTINUE
C BEREGNING AV KOEFFISIENTMATRISE FOR STEDET X,Y-----
DO 351 I=1,IORD
AAASUM=0.0
DO 352 J=1,IKURV
BBB=DMAT(I,J)
CCC=ZVEKT(J)
AAASUM=AAASUM+BBB*CCC
352 CONTINUE
DVEKT(I)=AAASUM
351 CONTINUE
C BEREGNING AV COVARIANSMATRISE FOR STEDET X,Y-----
IF(IKURV-2) 360,370,380
C EN KURVE-----
360 CALL LFEED(2)
IGRAD=IREAD(' Gi usikkerhetsgradient: ',25)
ZVEKT(1)=1+IGRAD*SQRT((XKOORD-X(1))**2+(YKOORD-Y(1))**2)
GOTO 380
C TO KURVER-----
370 DIST=(2*XKOORD-X(1)-X(2))**2+(2*YKOORD-Y(1)-Y(2))**2
DIST=SQRT(DIST/FKONST)
ZVEKT(1)=0.5*(1-DIST)
ZVEKT(2)=0.5*(1+DIST)
C FLERE KURVER-----

```

```
74 FORMAT(1X,10E15.9)
71 CONTINUE
ENDFILE 9
GOTO 10
C SLUTT-----
END
```

```
IF(ILOG.EQ.1) ZNO2=ALOG(HOYD2+1)/ALOG(ZHL+1)
TID1=TIDCAL(ZNO1,TIDMIN,IORD,DVEKT)
TID2=TIDCAL(ZNO2,TIDMIN,IORD,DVEKT)
TID1=TID0-TID1
TID2=TID2-TID0
TIDM=(TID1+TID2)/2
IT=TID0/10
TID0=10*IT
IT=TIDM/10
TIDM=10*IT
WRITE(3,59) TID0,TIDM
59 FORMAT(' Tidspunktet er: ',F6.0,' +- ',F5.0,' år BP////////)
ITID=(TID0+2000)/1000
TIDMIN=ITID*1000
GOTO 51
```

```
C PLOTTING-----
60 CALL OPEN(8,'KURVMAT TMP',0)
CALL OPEN(7,'KURVDAT TMP',0)
WRITE(3,65)
65 FORMAT(' Gi navn på stedet: ')
READ(3,66) (NAVN(K),K=1,15)
66 FORMAT(15A1)
WRITE(7,63)
63 FORMAT(1X)
XKOORD=1000.*XKOORD+1000.
YKOORD=1000.*YKOORD+6000.
TTHL=10600.
WRITE(8,67) IORD,ILOG
67 FORMAT(1X,2I1)
WRITE(8,64) XKOORD,YKOORD,(NAVN(K),K=1,15),TTHL,ZHL
64 FORMAT(1X,2F5.0,1X,15A1,1X,F6.0,F4.0)
DO 61 I=1,7
WRITE(8,62) DVEKT(I),(COV(I,K),K=1,7)
62 FORMAT(1X,8E15.9)
61 CONTINUE
ENDFILE 7
ENDFILE 8
CALL LFEED(2)
CALL FCHAIN('KURVPLYCOM',0)
C UTSKRIFT AV E-MATRICE-----
70 CALL FIREAD(EFILE,'E?????.MAT')
```

```
CALL OPEN(9,EFILE,0)
WRITE(9,75) IORD,IKURV
DO 71 I=1,IORD
DO 72 K=1,IKURV1
AAASUM=0.0
DO 73 J=1,IKURV
BBB=DMAT(I,J)
CCC=SINV(K,J)
AAAASUM=AAAASUM+BBB*CCC
73 CONTINUE
E(I,K)=AAAASUM
72 CONTINUE
WRITE(9,74) (E(I,K),K=1,IKURV1)
WRITE(3,74) (E(I,K),K=1,IKURV1)
```

```

C*****
C
C PROGRAM FOR BEREGNING AV FLATEHØYDER
C
C Programert av Kjell Kjenstad, UiO
C Desember 1982
C
C*****
PROGRAM FLATBE
BYTE HFILE(11)
REAL H(5,5)
WRITE(3,101)
101 FORMAT(1X,//////////)
* ,8X, '*****'
* ,8X, '***'
* ,8X, '***'
* ,8X, '***'
* ,8X, '***'
* ,8X, '***'
CALL LFEED(10)
C INNLESNING AV H-MATRISER-----
CALL FIREAD(HFILE, '??XX? .MAT')
CALL OPEN(7,HFILE,0)
READ(7,21) IHORD
21 FORMAT(I15)
IH=IHORD+1
DO 22 I=1,IH
READ(7,23) H(I,J),J=1,IH
23 FORMAT(6F15.9)
22 CONTINUE
ENDFILE 7
CALL LFEED(30)
C INNLESNING AV STED-----
30 IKOORD=IREAD(' Gi X-koordinaten til stedet: ',30)
IF(IKOORD.LE.0) CALL FCHAIN('MODELL COM',0)
JKOORD=IREAD(' Gi Y-koordinaten til stedet: ',30)
XKOORD=JKOORD
YKOORD=JKOORD
XKOORD=XKOORD/1000.
YKOORD=YKOORD/1000.
C BEREGNING AV HOVEDFLATA FOR STEDET X Y -----
ZZHL=ZHCAL(XKOORD,YKOORD,IHORD,H)
WRITE(3,31) ZZHL
31 FORMAT(/' Hovedlinjehøyden er: ',F7.2////////)
C SLUTT-----
END

```

```

C*****
C
C Program for gradientberegninger på havnivåflater
C
C Programert av Kjell Kjenstad, UiO
C Januar 1983
C
C*****
PROGRAM GRADIE
BYTE HFILE(11),EFILE(11)
REAL EMAT(7,10),DVEKT(7),HMAT(5,5),SVEKT(15)
* ,X(2),Y(2),Z(2,2)
DOUBLE PRECISION AAASUM,BBBSUM,BBB,CCC,DDD
DATA EFILE/'E',2*X,5*' ',M',A','T'/'
DATA HFILE/'H','L',2*X,4*' ',M',A','T'/'
C HEADING-----
WRITE(3,12)
12 FORMAT(1X,//////////)
* ,8X, '*****'
* ,8X, '***'
* ,8X, '***'
* ,8X, '***'
* ,8X, '***'
* ,8X, '***'
* ,8X, '*****'
*/
CALL LFEED(5)
C PARAMETERDIALOG-----
20 IBAS=IREAD(' Gi databasenummer: ',20)
ILOG=LREAD(' Logaritmetransformasjon (Ja)',29)
C INNLESNING AV E-MATRISERFIL-----
EFILE(4)=48+IBAS
CALL OPEN(9,EFILE,0)
READ(9,24,ERR=98,END=99) IORD,IKURV
24 FORMAT(2I1)
IKURV=IKURV
IF(IKURV.EQ.2) IKURV=3
DO 21 I=1,IORD
READ(9,22,ERR=98,END=99) (EMAT(I,K),K=1,IKURV)
22 FORMAT(10E15.9)
WRITE(3,23) (EMAT(I,K),K=1,IKURV)
23 FORMAT(1X,10E15.9)
21 CONTINUE
ENDFILE 9
CALL LFEED(3)
C INNLESNING AV H-MATRISERFIL-----
HFILE(5)=48+IBAS
CALL OPEN(7,HFILE,0)
READ(7,25,ERR=98,END=99) IHORD
25 FORMAT(I15)
IH=IHORD+1
DO 26 I=1,IH
READ(7,27,ERR=98,END=99) (HMAT(I,J),J=1,IH)
WRITE(3,28) (HMAT(I,J),J=1,IH)
27 FORMAT(6F15.9)
28 FORMAT(1X,6E15.9)
26 CONTINUE

```

```
ENDFILE 7
CALL LFEED(3)
30 CALL LFEED(5)
C HØYDEBEREGNINGSLØSKE-----
IKOORD=IREAD(' Gi x-koordinat: ',18)
IF(IKOORD.EQ.0) CALL FCHAIN('MODELL COM',0)
JKOORD=IREAD(' Gi y-koordinat: ',18)
XKOORD=IKOORD
YKOORD=JKOORD
XKOORD=XKOORD/1000
YKOORD=(YKOORD-6000)/1000
C BEREGNING AV T-VEKTOR-----
40 CALL LFEED(3)
ITID=IREAD(' Gi tidspunktet: ',18)
IF(ITID.LE.0) GOTO 30
TID=ITID
C BEREGNING AV STEDMATRISE FOR PUNKTET X,Y-----
DO 31 II=1,2
DO 31 JJ=1,2
X(II)=XKOORD+(II-1)*0.001
Y(JJ)=YKOORD+(JJ-1)*0.001
SVEKT(1)=1.
SVEKT(2)=X(II)
SVEKT(3)=Y(JJ)
SVEKT(4)=X(II)*X(II)
SVEKT(5)=X(II)*Y(JJ)
SVEKT(6)=Y(JJ)*Y(JJ)
SVEKT(7)=X(II)*X(II)*X(II)
SVEKT(8)=X(II)*X(II)*Y(JJ)
SVEKT(9)=X(II)*Y(JJ)*Y(JJ)
SVEKT(10)=Y(JJ)*Y(JJ)*Y(JJ)
C BEREGNING AV KOFFISIENTMATRISE FOR STEDET X,Y-----
DO 41 I=1,IORD
AAASUM=0.0
DO 42 K=1,IKURV1
BBB=EMAT(I,K)
CCC=SVEKT(K)
AAASUM=AAASUM+BBB*CCC
42 CONTINUE
DVEKT(I)=AAASUM
41 CONTINUE
C BEREGNING AV HOVEDFLATA FOR STEDET X Y -----
ZHL=ZHLCAL(X(II),Y(JJ),IHORD,HMAT)
C BEREGNING AV HAVNIVÅHØYDE PÅ STEDET X Y-----
ZNO=HOYCAL(TID,IORD,DVEKT)
Z(II,JJ)=EXP(ZNO*ALOG(ZHL+1))-1
IF((II.NE.1).OR.(JJ.NE.1)) GOTO 31
WRITE(3,43) ZHL,ZNO,Z(II,JJ)
43 FORMAT(' Hovedlinje: ',F5.1,' Normalhøyde: ',F5.3,
* ' Høyde: ',F6.1)
31 CONTINUE
C GRADIENTBEREGNING-----
DX=Z(2,1)-Z(1,1)
DY=Z(1,2)-Z(1,1)
GRAD=SQRT(DX*DX+DY*DY)
TETA=ATAN2(DX,DY)*(180/3.141592)
IF(TETA.LT.0) TETA=180.+TETA
WRITE(3,71) GRAD,TETA
71 FORMAT('/ Gradient: ',F6.2,' Vinkel: ',F5.1/)
GOTO 40
C END-----
98 CALL TWRITE('***** READ-ERROR *****',25)
STOP
99 CALL TWRITE('***** END-OF-FILE *****',25)
STOP
END
```

```

C*****
C Program under PALEKART for datering av MG-lokaliteter
C
C Programert av Kjell Kjenstad, Uio
C Desember 1982
C
C*****
PROGRAM MGCALC
BYTE EFILE(11),HFILE(11),FILE(11),NAVN(15),DANAVN(15)
*,FORF(3)
REAL DMAT(7,10),SARR(10,10),COVARR(7,7,10),X(10),Y(10),ZHL(10)
*,SINV(10,10),ZVEKT(10),SVEKT(10),DVEKT(7),COV(7,7),H(5,5),E(7,10)
DOUBLE PRECISION AAASUM,BBSUM,AAAHOF,AAADIF,BBB,CCC,DDD
INTEGER IVEKT(10)
DATA MFILE/'M','G','2','X','4',' ','D',' ','A','T'/'
C HEADING-----
WRITE(3,10)
10 FORMAT(1X,//////////)
*,8X, '*****'
*,8X, '***'
*,8X, '***'
*,8X, '***'
*,8X, '***'
*,8X, '***'
*,8X, '*****'
*/
C INNLESNING AV TEMPORÆRFIL-----
CALL OPEN(9,'KURNAV TMP',0)
CALL TWRITE(' Tempørærfilen består av følgende data: ',40)
CALL LFEED(2)
CALL TWRITE(' Matrisefil for Hovedlinjehøyde: ',40)
READ(9,3,ERR=8,END=8) (HFILE(K),K=1,11)
WRITE(3,3)(HFILE(K),K=1,11)
CALL LFEED(1)
READ(9,1,ERR=8,END=8) IKURV,IORD
1 FORMAT(2I1)
WRITE(3,7) IKURV,IORD
7 FORMAT(' Modellen består av ',I1,' kurver av '
*/IX,I1,' orden fra følgende matrisefiler:/'
DO 2 II=1,IKURV
READ(9,3,ERR=8,END=8) (FILE(K),K=1,11)
3 FORMAT(1X,11A1)
CALL LFEED(1)
CALL OPEN(6,FILE,0)
READ(6,4,ERR=9,END=9) X(II),Y(II),(NAVN(K),K=1,15),THL,ZHL(II)
WRITE(3,4) X(II),Y(II),(NAVN(K),K=1,15),THL,ZHL(II)
4 FORMAT(2F5.0,1X,15A1,1X,F6.0,F4.0)
X(II)=(X(II)-1000.)/1000.
Y(II)=(Y(II)-6000.)/1000.
DO 5 I=1,IORD
READ(6,6) DMAT(I,II),(COVARR(I,K,II),K=1,7)
6 FORMAT(8E15.9)
5 CONTINUE
9 ENDFILE 6
CALL LFEED(1)
2 CONTINUE
C *****
8 ENDFILE 9
C INVERTING AV STEDMATRISE-----
DO 11 I=1,10
DO 11 J=1,10
SINV(I,J)=0.0
11 CONTINUE
IF(IKURV-2) 12,13,14
C 1 KURVE-----
12 SINV(1,1)=1.0
IKURV1=1
GOTO 20
C 2 KURVER-----
13 FKONST=(X(1)-X(2))*2+(Y(1)-Y(2))*2
SINV(1,1)=(X(2)-X(1))*X(2)+(Y(2)-Y(1))*Y(2)/FKONST
SINV(1,2)=(X(1)-X(2))*X(1)+(Y(1)-Y(2))*Y(1)/FKONST
SINV(1,3)=0.
SINV(2,1)=(X(1)-X(2))/FKONST
SINV(2,2)=(X(2)-X(1))/FKONST
SINV(2,3)=0.
SINV(3,1)=(Y(1)-Y(2))/FKONST
SINV(3,2)=(Y(2)-Y(1))/FKONST
SINV(3,3)=0.
IKURV1=3
GOTO 20
C FLERE KURVER-----
14 DO 15 I=1,IKURV
SARR(I,1)=1.
SARR(I,2)=X(I)
SARR(I,3)=Y(I)
SARR(I,4)=X(I)*X(I)
SARR(I,5)=X(I)*Y(I)
SARR(I,6)=Y(I)*Y(I)
SARR(I,7)=X(I)*X(I)*X(I)
SARR(I,8)=X(I)*X(I)*Y(I)
SARR(I,9)=X(I)*Y(I)*Y(I)
SARR(I,10)=Y(I)*Y(I)*Y(I)
15 CONTINUE
IKURV1=IKURV
CALL INVERS(SARR,10,IKURV,SINV,ZVEKT,IVEKT,IBERR)
C INNLESNING AV HL-FLATE-----
20 CALL OPEN(7,HFILE,0)
READ(7,25) IHORD
25 FORMAT(I15)
IH=IHORD+1
DO 26 I=1,IH
READ(7,27) (H(I,J),J=1,IH)
27 FORMAT(6F15.9)
26 CONTINUE
ENDFILE 7
C ÅPNING AV MG-LOKALITETS-FIL-----
CALL LFEED(2)
MFILE(5)=48+IREAD(' Gi databasennummer: ',20)
CALL LFEED(2)
WRITE(3,28) (MFILE(K),K=1,11)
28 FORMAT(' Innputfil: ',8A1,'.',3A1)
CALL LFEED(1)

```

```

CALL OPEN(7,MFILE,0)
MFILE(3)='Y'
MFILE(4)='Y'
WRITE(3,29) (MFILE(K),K=1,11)
29 FORMAT(' Outputfil: ',8A1,' ',3A1)
CALL OPEN(8,MFILE,0)
CALL LFEED(2)
C PARAMETERDIALOG-----
ISD=LREAD(' Logaritmetransformasjon (Ja)',29)
ZHLVAR=ISD*ISD
IGRAD=IREAD(' Gi usikkerhetsgradient: ',25)
IANT=IREAD(' Gi minste nummerverdi: ',24)
ITMI=IREAD(' Gi minste årstall : ',24)
ITWA=IREAD(' Gi største årstall : ',24)
C LESELØKKE-----
30 IANT=IANT+1
READ(7,31,ERR=90,END=90) XX,YY,(NAVN(K),K=1,15),ZZ
*,(FORF(K),K=1,3)
31 FORMAT(2F5.0,1X,15A1,1X,F4.0,5X,3A1)
HOYD=ZZ
XKOORD=(XX-1000.)/1000.
YKOORD=(YY-6000.)/1000.
DO 341 I=1,10
SVEKT(I)=0.0
341 CONTINUE
DO 342 I=1,7
DVEKT(I)=0.0
DO 342 J=1,7
COV(I,J)=0.0
342 CONTINUE
SVEKT(1)=1.
SVEKT(2)=XKOORD
SVEKT(3)=YKOORD
SVEKT(4)=XKOORD*XKOORD
SVEKT(5)=XKOORD*YKOORD
SVEKT(6)=YKOORD*YKOORD
SVEKT(7)=XKOORD*XKOORD*XKOORD
SVEKT(8)=XKOORD*XKOORD*YKOORD
SVEKT(9)=XKOORD*YKOORD*YKOORD
SVEKT(10)=YKOORD*YKOORD*YKOORD
DO 343 J=1, IKURV
AAASUM=0.0
DO 344 K=1, IKURV1
CCC=SINV(K,J)
DDD=SVEKT(K)
AAASUM=AAASUM+CCC*DDD
344 CONTINUE
ZVEKT(J)=AAASUM
343 CONTINUE
DO 351 I=1, IORD
AAASUM=0.0
DO 352 J=1, IKURV
BBB=DMAT(I,J)
CCC=ZVEKT(J)

```

```

AAASUM=AAASUM+BBB*CCC
352 CONTINUE
DVEKT(I)=AAASUM
351 CONTINUE
C BEREGNING AV COVARIANSMATRISE FOR STEDET X,Y-----
C EN KURVE-----
360 ZVEKT(1)=1+IGRAD*(SQRT((XKOORD-X(1))*2+(YKOORD-Y(1))*2))
GOTO 380
C TO KURVER-----
370 DIST=(2*XKOORD-X(1)-X(2))*2+(2*YKOORD-Y(1)-Y(2))*2
DIST=SQRT(DIST/FKONST)
ZVEKT(1)=0.5*(1-DIST)
ZVEKT(2)=0.5*(1+DIST)
C FLERE KURVER-----
380 DO 381 I=1, IORD
DO 381 J=1, IORD
AAASUM=0.0
DO 382 K=1, IKURV
BBB=COVARR(I,J,K)
CCC=ZVEKT(K)
AAASUM=AAASUM+BBB*CCC*CCC
382 CONTINUE
COV(I,J)=AAASUM
381 CONTINUE
C BEREGNING AV HOVEDFLATA FOR STEDET X Y -----
ZZHL=ZHLCAL(XKOORD,YKOORD,IHORD,H)
C TIDSPUNKTSBEREGNING-----
50 HOYD=ZZ
TIDMIN=1000*(ITMI/1000)+1000
IF(ILOG.EQ.0) ZNO=HOYD/ZZHL
IF(ILOG.EQ.1) ZNO=ALOG(HOYD+1)/ALOG(ZZHL+1)
TID0=TIDCAL(ZNO,TIDMIN,IORD,DVEKT)
IF(TID0.GE.ITWA) GOTO 30
TID=TID0
ZNOVAR=VARCAL(TID,IORD,COV)
IF(ILOG.EQ.0) CALL LINTRA(HOYD,HOYD1,HOYD2,ZZHL,ZHLVAR,ZNO,ZNOVAR)
IF(ILOG.EQ.1) CALL LOGTRA(HOYD,HOYD1,HOYD2,ZZHL,ZHLVAR,ZNO,ZNOVAR)
IF(ILOG.EQ.0) ZNO1=HOYD1/ZZHL
IF(ILOG.EQ.1) ZNO1=ALOG(HOYD1+1)/ALOG(ZZHL+1)
IF(ILOG.EQ.0) ZNO2=HOYD2/ZZHL
IF(ILOG.EQ.1) ZNO2=ALOG(HOYD2+1)/ALOG(ZZHL+1)
TID1=TIDCAL(ZNO1,TIDMIN-1000,IORD,DVEKT)
TID2=TIDCAL(ZNO2,TIDMIN,IORD,DVEKT)
TID1=ABS(TID0-TID1)
TID2=ABS(TID2-TID0)
TIDM=AMINI(TID1,TID2)
IT=TID0/10
TID0=10*IT
IT=TIDM/10
TIDM=10*IT
IF(TIDM.GE.1000.) TIDM=999
IF(TIDM.LE.-100.) TIDM=999
IF(TID0.LE.ITMI) GOTO 30
WRITE(8,52) IANT,(NAVN(K),K=1,15),XX,YY,ZZ,ZZHL,TID0,TIDM
*,(FORF(K),K=1,3)

```

```
52 FORMAT(1X,I3,1X,15A1,2F6.0,2F5.0,F7.0,'+-',F4.0,1X,3A1)
   WRITE(3,53) IANT,(NAVN(K),K=1,15),XX,YY,ZZ,ZZHL,TID0,TIDM
   *,(FORF(K),K=1,3)
53 FORMAT(1X,I3,1X,15A1,2F6.0,2F5.0,F7.0,'+-',F4.0,1X,3A1)
   GOTO 30
C SLUTT-----
90 ENDFILE 7
ENDFILE 8
CALL FCHAIN('MENYPRYCOM',0)
C SLUTT-----
END
```



```

*****
**
**
**
*****
KARTPLOTTEPROGRAMMER
*****
MENYPRZZ: er meny-programmet for
kartplotteprogrampakken.....S. A - 33

INITIAZZ: foretar definering av kartparametre og
formatering av den grafiske hukommelsen.....S. A - 34

TOPPUTZZ: foretar innlesning av de topografiske
dataene fra diskfil.....S. A - 36

KARPUTZZ: foretar utregning av kart over dagens
topografi.....S. A - 36

LOKALIZZ: foretar utplotting av lokaliteter fra
en av datalistene.....S. A - 37

FLALESZZ: foretar innlesning av høydene på en
parametrisert flate.....S. A - 38

HAVLESZZ: foretar innlesning av høydene på
havnivå for et gitt tidspunkt.....S. A - 39

TIDLESZZ: foretar innlesning av isavsmeltings-
tidspunkt.....S. A - 41

RUTMIDZZ: foretar glatting av tema i 10 x 10
ruter.....S. A - 42

HAVTOPZZ: foretar uttegning av et paleogeografisk
kart uten isbre.....S. A - 43

ISOBASZZ: foretar uttegning av et isolinjekart.....S. A - 44

MARINEZZ: foretar uttegning av marine områder.....S. A - 46

KARPALZZ: foretar en transformasjon til topo-
grafiske høydeverdier for et valgt
tidspunkt.....S. A - 47

ISFLATZZ: foretar beregning av iskroppens
overflateform.....S. A - 48

ISFMIDZZ: foretar glatting av isflaten.....S. A - 50

ISKARTZZ: foretar plotting av et paleogeografisk
kart.....S. A - 51

*****
C Menyprogram for PALEOKART- programmer
C
C Programmert av Kjell Kjenstad, UiO
C Januar 1983
C
C*****
PROGRAM MENYZZ
11 CALL TDV(25)
WRITE(3,12)
12 FORMAT(
*,8X,'***** Hovedmeny: Beregning og plotting av kart
*,8X,'**
*,8X,'**
*,8X,'**
*,8X,'**
*,8X,'**
*,8X,'**
*,8X,'**
*,8X,'**
*,8X,'**
*,8X,'**
*,8X,'**
*,8X,'**
*,8X,'**
*,8X,'**
*,8X,'**
*,8X,'**
*,8X,'**
*,8X,'**
*,8X,'**
*,8X,'**
*,8X,'**
*/
*,8X,'*****
*/
)
READ(3,13) NEXT
13 FORMAT(I2)
CALL TDV(25)
IF(NEXT.LE.0) STOP
IF(NEXT.GT.13) GOTO 11
GOTO (10,20,30,40,50,60,70,80,90,100,110,120,130).NEXT
C-----
10 CALL FCHAIN('INITIAZZCOM',0)
20 CALL FCHAIN('TOPPUTZZCOM',0)
30 CALL FCHAIN('KARPUTZZCOM',0)
40 CALL FCHAIN('LOKALIZZCOM',0)
50 CALL FCHAIN('FLALESZZCOM',0)
60 CALL FCHAIN('HAVLESZZCOM',0)
70 CALL FCHAIN('TIDLESZZCOM',0)
80 CALL FCHAIN('RUTMIDZZCOM',0)
90 CALL FCHAIN('HAVTOPZZCOM',0)
100 CALL FCHAIN('ISOBASZZCOM',0)
110 CALL FCHAIN('MARINEZZCOM',0)
120 CALL FCHAIN('KARPALZZCOM',0)
130 CALL FCHAIN('ISPALEZZCOM',0)
END

```

```

*****
**
**
**
*****
KARTPLOTTEPROGRAMMER
*****
MENYPRZZ: er meny-programmet for
kartplotteprogrampakken.....S. A - 33

INITIAZZ: foretar definering av kartparametre og
formatering av den grafiske hukommelsen.....S. A - 34

TOPPUTZZ: foretar innlesning av de topografiske
dataene fra diskfil.....S. A - 36

KARPUTZZ: foretar utregning av kart over dagens
topografi.....S. A - 36

LOKALIZZ: foretar utplotting av lokaliteter fra
en av datalistene.....S. A - 37

FLALESZZ: foretar innlesning av høydene på en
parametrisert flate.....S. A - 38

HAVLESZZ: foretar innlesning av høydene på
havnivå for et gitt tidspunkt.....S. A - 39

TIDLESZZ: foretar innlesning av isavsmeltings-
tidspunkt.....S. A - 41

RUTMIDZZ: foretar glatting av tema i 10 x 10
ruter.....S. A - 42

HAVTOPZZ: foretar uttegning av et paleogeografisk
kart uten isbre.....S. A - 43

ISOBASZZ: foretar uttegning av et isolinjekart.....S. A - 44

MARINEZZ: foretar uttegning av marine områder.....S. A - 46

KARPALZZ: foretar en transformasjon til topo-
grafiske høydeverdier for et valgt
tidspunkt.....S. A - 47

ISFLATZZ: foretar beregning av iskroppens
overflateform.....S. A - 48

ISFMIDZZ: foretar glatting av isflaten.....S. A - 50

ISKARTZZ: foretar plotting av et paleogeografisk
kart.....S. A - 51

*****
C Menyprogram for PALEOKART- programmer
C
C Programmert av Kjell Kjenstad, UiO
C Januar 1983
C
C*****
PROGRAM MENYZZ
11 CALL TDV(25)
WRITE(3,12)
12 FORMAT(
*,8X,'***** Hovedmeny: Beregning og plotting av kart
*,8X,'**
*,8X,'**
*,8X,'**
*,8X,'**
*,8X,'**
*,8X,'**
*,8X,'**
*,8X,'**
*,8X,'**
*,8X,'**
*,8X,'**
*,8X,'**
*,8X,'**
*,8X,'**
*,8X,'**
*,8X,'**
*,8X,'**
*,8X,'**
*,8X,'**
*,8X,'**
*,8X,'**
*/
*,8X,'*****
*/
)
READ(3,13) NEXT
13 FORMAT(I2)
CALL TDV(25)
IF(NEXT.LE.0) STOP
IF(NEXT.GT.13) GOTO 11
GOTO (10,20,30,40,50,60,70,80,90,100,110,120,130).NEXT
C-----
10 CALL FCHAIN('INITIAZZCOM',0)
20 CALL FCHAIN('TOPPUTZZCOM',0)
30 CALL FCHAIN('KARPUTZZCOM',0)
40 CALL FCHAIN('LOKALIZZCOM',0)
50 CALL FCHAIN('FLALESZZCOM',0)
60 CALL FCHAIN('HAVLESZZCOM',0)
70 CALL FCHAIN('TIDLESZZCOM',0)
80 CALL FCHAIN('RUTMIDZZCOM',0)
90 CALL FCHAIN('HAVTOPZZCOM',0)
100 CALL FCHAIN('ISOBASZZCOM',0)
110 CALL FCHAIN('MARINEZZCOM',0)
120 CALL FCHAIN('KARPALZZCOM',0)
130 CALL FCHAIN('ISPALEZZCOM',0)
END

```



```
IF((NRUTE.EQ.3).OR.(NRUTE.EQ.7)) CALL XAREA(18,121,198,241,0)
IF((NRUTE.EQ.4).OR.(NRUTE.EQ.8)) CALL XAREA(199,121,378,241,0)
GOTO 40
C RUTEFORANDRING-----
50 CALL DEKODE
CALL PARAME
KPRUTE=IREAD(' Hvilken rute skal plottes: ',28)
IF(KPRUTE.EQ.0) GOTO 50
JY(6)=(KPRUTE-1)/2
IX(6)=(KPRUTE-2*JY(6))-1
51 KARUTE=IREAD(' Hvilken rute skal behandles: ',30)
IF(KARUTE.EQ.0) GOTO 51
JY(7)=(KARUTE-1)/2
IX(7)=(KARUTE-2*JY(7))-1
CALL ENKODE
CALL PARAME
GOTO 1
C KARTPARAMETRE-----
60 CALL DEKODE
IBASE=IREAD(' Gi topbasenummer: ',20)
FILE(5)=IBASE+48
CALL OPEN(6,FILE,0)
READ(6,REC=1,ERR=90) IBASEL,IOBSL,IMIN1,IMIN2,JMIN1,JMIN2
*,IMAX1,IMAX2,JMAX1,JMAX2
ENDFILE 6
IOBS=IOBSL
IX(1)=100*IMIN1+IMIN2
JY(1)=100*JMIN1+JMIN2
IX(5)=100*IMAX1+IMAX2
JY(5)=100*JMAX1+JMAX2
CALL PARAME
IX(2)=IREAD(' Gi x-origo: ',13)
JY(2)=IREAD(' Gi y-origo: ',13)
IRANGE=IREAD(' Gi x-range: ',13)
JRANGE=IREAD(' Gi y-range: ',13)
C -----X-KOORDINATER-----
IF(IRANGE.LE.0) IRANGE=36*IOBS
IF(IRANGE.GT.(36*IOBS)) IRANGE=36*IOBS
IF(IX(2).LT.IX(1)) IX(2)=IX(1)
IX(4)=IX(2)+JRANGE
IF(IX(4).GT.IX(5)) IX(4)=IX(5)
IX(3)=IX(2)+18*IOBS
IF(IX(3).GT.IX(4)) IX(3)=IX(4)
C -----Y-KOORDINATER-----
IF(JRANGE.LE.0) JRANGE=24*IOBS
IF(JRANGE.GT.(24*IOBS)) JRANGE=24*IOBS
IF(JY(2).LT.JY(1)) JY(2)=JY(1)
JY(4)=JY(2)+JRANGE
IF(JY(4).GT.JY(5)) JY(4)=JY(5)
JY(3)=JY(2)+12*IOBS
IF(JY(3).GT.JY(4)) JY(3)=JY(4)
CALL ENKODE
CALL PARAME
GOTO 1
C FORANDRING AV PAKNINGSGRENSER-----
70 CALL DEKODE
```

```
CALL PARAME
IHMIN=IREAD(' Gi minste pakningsgrense: ',27)
IHMAX=IREAD(' Gi største pakningsgrense: ',28)
CALL ENKODE
CALL PARAME
GOTO 1
C READERROR-----
90 CALL TWRITE(' ***** READ-ERROR *****',27)
STOP
END
```

```
C*****
C
C Program under PALEKART for lesing fra database til skjerm 1
C
C Programmert av Kjell Kjenstad, UiO
C Mai 1983
C
C*****
PROGRAM TOPPUT
BYTE FILE(11)
INTEGER ITOPO(10,10)
COMMON IPAGE,IBASE,IOBS,IHFAKT,IHMIN,IX(7),JY(7)
DATA FILE/'B','A','S','E',4*','D','A','T'/'
C SDI-----
CALL GRAFIX
IPAGE=1
CALL SDIINI
CALL MAP4
C HEADING-----
CALL TDV(25)
WRITE(3,12)
12 FORMAT(1X,/,
*,8X,*****
*,8X,**,
*,8X,**,
*,8X,**,
*,8X,**,
*,8X,*****
*)
PROGRAM for lesing fra topobase
til skjerm
*,8X,**,
*,8X,*****
*)
C LESING AV PARAMETRE-----
CALL DEKODE
CALL PARAME
C BEREGNING AV PARAMETRE-----
IARUTE=IX(7)
JARUTE=JY(7)
IPRUTE=IX(6)
JPRUTE=JY(6)
IORIGO=IX(IARUTE+2)
JORIGO=JY(IARUTE+2)
IRANGE=(IX(IARUTE+3)-IX(IARUTE+2))*IOBS/10
JRANGE=(JY(JARUTE+3)-JY(JARUTE+2))*IOBS/10
CALL EKVIDI(IEKV,IEKVM1)
IA=IEKV-IEKVM1-1
B=IEKV
B=1/B
C TEGNE-LØKKE-----
DO 21 J=1,JRANGE
DO 21 I=1,IRANGE
CALL WORKON(I)
ITOP=10*INTGET(I,J,0)
ITOP=ITOP+IA
ITOP=ITOP*B
IF(ITOP.LT.0) ITOP=0
IF(ITOP.GT.7) ITOP=7
CALL WORKON(0)
CALL BYTPUT(I,J,IPRUTE,JPRUTE,ITOP)
21 CONTINUE
CALL TDV(25)
CALL FCHAIN('MENYPRZZCOM',0)
END

C*****
C
C Program under PALEKART for tegning av topografisk kart
C
C Programmert av Kjell Kjenstad, UiO
C Mai 1983
C
C*****
PROGRAM KARPUP
COMMON IPAGE,IBASE,IOBS,IHMAX,IHMIN,IX(7),JY(7)
C SDI-----
CALL GRAFIX
IPAGE=0
CALL SDIINI
CALL MAP4
C HEADING-----
CALL TDV(25)
WRITE(3,12)
12 FORMAT(1X,/,
*,8X,*****
*,8X,**,
*,8X,**,
*,8X,**,
*,8X,**,
*,8X,*****
*)
PROGRAM for uttegnig av
topografisk kart
*,8X,**,
*,8X,*****
*)
C LESING AV PARAMETRE-----
CALL DEKODE
CALL PARAME
C BEREGNING AV PARAMETRE-----
IARUTE=IX(7)
JARUTE=JY(7)
IPRUTE=IX(6)
JPRUTE=JY(6)
IORIGO=IX(IARUTE+2)
JORIGO=JY(IARUTE+2)
IRANGE=(IX(IARUTE+3)-IX(IARUTE+2))*IOBS/10
JRANGE=(JY(JARUTE+3)-JY(JARUTE+2))*IOBS/10
CALL EKVIDI(IEKV,IEKVM1)
IA=IEKV-IEKVM1-1
B=IEKV
B=1/B
C TEGNE-LØKKE-----
DO 21 J=1,JRANGE
DO 21 I=1,IRANGE
CALL WORKON(I)
ITOP=10*INTGET(I,J,0)
ITOP=ITOP+IA
ITOP=ITOP*B
IF(ITOP.LT.0) ITOP=0
IF(ITOP.GT.7) ITOP=7
CALL WORKON(0)
CALL BYTPUT(I,J,IPRUTE,JPRUTE,ITOP)
21 CONTINUE
CALL TDV(25)
CALL FCHAIN('MENYPRZZCOM',0)
END
```

```
C*****
C
C Program under PALEKART for plotting av lokaliteter
C
C Programert av Kjell Kjenstad, UiO
C Januar 1983
C
C*****
PROGRAM MGLOKA
BYTE FILE(11)
COMMON IPAGE,IBASE,IOBS,IHMAX,IHMIN,IX(7),JY(7)
DATA FILE/2' ','2*X','4' ','D','A','T'/'
C SDI INITIALISERING-----
CALL GRAFIX
IPAGE=0
CALL SDINI
CALL MAP4
C HEADING-----
CALL TDV(25)
WRITE(3,12)
12 FORMAT(1X, /
*,8X,'*****'
*,8X,'**
*,8X,'**
*,8X,'**
*,8X,'**
*,8X,'**
*,8X,'*****'
C LESING AV PARAMETRE-----
CALL DEKODE
CALL PARAME
C FIL-ÅPNING-----
CALL TDV(25)
WRITE(3,21)
21 FORMAT(' Gi filtypesifikasjon (HL,MG,etc): ')
22 FORMAT(2A1)
FILE(5)=48+IREAD(' Gi databasenummer: ',20)
CALL OPEN(6,FILE,0)
C PARAMETERBEREGNING-----
40 IARUTE=IX(7)
JARUTE=JY(7)
IPRUTE=IX(6)
JPRUTE=JY(6)
IORIGO=IX(IARUTE+2)
JORIGO=JY(JARUTE+2)
C PLOTTELØKKE-----
DO 42 IA=1,500
READ(6,43,ERR=99,END=44) X,Y
43 FORMAT(2F5.0)
I=X-IORIGO-IBASE*1000
J=Y-JORIGO
IF((I.LE.0).OR.(I.GE.180).OR.(J.LE.0).OR.(J.GE.120)) GOTO 42
CALL BYTPUT(I,J,IPRUTE,JPRUTE,15)
42 CONTINUE
44 ENDFILE 6
C SLUTT-----
```

```
CALL TDV(25)
CALL FCHAIN('MENYPRZCOM',0)
C READ-ERROR-----
99 WRITE(3,98)
98 FORMAT(' ***** READ-ERROR ***** '/')
STOP
END
```

```

C*****
C
C Program under PALEKART for innlesning av flate
C
C Programmet av Kjell Kjenstad, Uio
C Januar 1983
C
C*****
PROGRAM FLALES
BYTE FILE(11)
REAL HMAT(5,5),FLATE(19,13)
INTEGER IFLATE(19,13),IKOORD(19)
COMMON IPAGE,IBASE,IOBS,IHMAX,IHMIN,IX(7),JY(7)
DATA FILE/2* ' ',2*X',4* ' ',M',A','T'/'
C SDI-----
CALL GRAFIX
IPAGE=1
CALL SDINI
CALL MAP4
C HEADING-----
CALL TDV(25)
WRITE(3,12)
12 FORMAT(
*,8X,'*****',
*,8X,'**',
*,8X,'**',
*,8X,'**',
*,8X,'**',
*,8X,'*****',
*/)
C LESING AV PARAMETRE-----
CALL DEKODE
CALL PARAME
C PARAMETERBEREGNING-----
IARUTE=IX(7)
JARUTE=JY(7)
IPRUTE=IX(6)
JPRUTE=JY(6)
IORIGO=IX(IARUTE+2)
JORIGO=JY(JARUTE+2)
IRANGE=((IX(IARUTE+3)-IX(IARUTE+2))*IOBS)/10
J RANGE=((JY(JARUTE+3)-JY(JARUTE+2))*IOBS)/10
IMAX1=IRANGE/10+1
JMAX1=J RANGE/10+1
IMAX2=IRANGE/10
JMAX2=J RANGE/10
IHMIN=30000
IHMAX=-30000
C FIL-ÅPNING-----
20 WRITE(3,21)
21 FORMAT(' Gi filtypesifikasjon (HL,NG,etc): ')
22 READ(3,22) FILE(1),FILE(2)
22 FORMAT(2A1)
FILE(5)=48+IREAD(' Gi databasenummer: ',20)
CALL OPEN(6,FILE,0)
C INNLESNING AV H-MATRISFIL-----
CALL OPEN(7,FILE,0)
CALL LFEED(1)
READ(7,25,ERR=98,END=99) IHORD
25 FORMAT(I15)
IH=IHORD+1
DO 26 I=1,IH
READ(7,27,ERR=98,END=99) (HMAT(I,J),J=1,IH)
27 FORMAT(6F15.9)
26 CONTINUE
ENDFILE 7
CALL LFEED(3)
C HØYDEBEREGNINGSLØSKE-----
DO 33 II=1,IMAX1
IKOORD(II)=IORIGO+IOBS*(II-1)
33 CONTINUE
WRITE(3,34) (IKOORD(II),II=1,IMAX1)
34 FORMAT(3X,19I4)
DO 31 J=1,JMAX1
JJ=JMAX1-J+1
DO 32 II=1,IMAX1
XKOORD=(JORIGO+10*(II-1))
YKOORD=(JORIGO+10*(JJ-1)-6000)
XKOORD=XKOORD/1000
YKOORD=YKOORD/1000
JKOORD=XKOORD/1000
JKOORD=YKOORD/1000
IF(JKOORD.GE.100) JKOORD=JKOORD-100
ZZHL=ZHLCAL(XKOORD,YKOORD,IHORD,HMAT)
IFLATE(II,JJ)=ZZHL
FLATE(II,JJ)=ZZHL
IHMAX=MAX0(IHMAX,IFLATE(II,JJ))
IHMIN=MIN0(IHMIN,IFLATE(II,JJ))
32 CONTINUE
WRITE(3,35) JKOORD,(IFLATE(K,JJ),K=1,IMAX1)
35 FORMAT(1X,I2,19(1X,I3))
31 CONTINUE
C BEREGNING AV PAKKEDE VERDIER-----
FKONST=255./(IHMAX-IHMIN)
WRITE(3,61) IHMIN
61 FORMAT('/ Minste verdi: ',I10)
WRITE(3,62) IHMAX
62 FORMAT(' Største verdi: ',I10/)
DO 63 I=1,IMAX1
DO 63 J=1,JMAX1
IFLATE(I,J)=FKONST*(FLATE(I,J)-IHMIN)
IF(IFLATE(I,J).GT.255) IFLATE(I,J)=255
63 CONTINUE
C PLOTTING-----
WRITE(3,71)
71 FORMAT(' 0: Utplotting' /
* ' 1: Nytt tidspunkt' /
* ' 2: Bare skrivning av parameterfil' )
CALL LFEED(1)
NEXT=IREAD(' : ',4)
CALL TDV(25)
IF(NEXT-1) 72,20,73

```

```
72 CALL RUTMAL(IFLATE)
73 CALL ENKODE
   CALL FCHAIN('MENYPRZZCOM',0)
C ERROR-LØKKE-----
98 CALL TWRITE('***** READ-ERROR *****',25)
99 CALL TWRITE('***** END-OF-FILE *****',25)
   STOP
   END

C*****
C Program under PALEKART for innlesning av havnivå
C
C Programert av Kjell Kjenstad, Uio
C Januar 1983
C*****
PROGRAM HAVLES
BYTE HFILE(11),EFILE(11)
REAL EMAT(7,10),SVEKT(10),DVEKT(7),HMAT(5,5),HAVET(19,13)
DOUBLE PRECISION AAASUM,AAAH0Y,BBB,CCC,DDD
INTEGER IHAVET(19,13),IKOORD(18)
COMMON IPAGE,IEASE,IOBS,IHMAX,IHMIN,IX(7),JY(7)
DATA EFILE/'E',2*'X',5*' ', 'M', 'A', 'T' /
DATA HFILE/'H', 'L', 2*'X', 4*' ', 'M', 'A', 'T' /

C SDI-----
CALL GRAFIX
IPAGE=1
CALL SDIINI
CALL MAP4

C HEADING-----
CALL TDV(25)
WRITE(3,12)
12 FORMAT(
*,8X,'*****
*,8X,'**
*,8X,'**
*,8X,'**
*,8X,'**
*,8X,'*****
C LESING AV PARAMETRE-----
CALL DEKODE
CALL PARAME
IARUTE=IX(7)
JARUTE=JY(7)
IPRUTE=IX(6)
JPRUTE=JY(6)
IORIGO=IX(IARUTE+2)
JORIGO=JY(JARUTE+2)
IRANGE=(IX(IARUTE+3)-IX(IARUTE+2))*IOBS/10
JRANGE=(JY(JARUTE+3)-JY(JARUTE+2))*IOBS/10

C PARAMETERDIALOG-----
20 IBAS=IREAD(' Gi databasenummer: ',20)
ILOG=LREAD(' Logaritmetransformasjon (Ja)',29)
ITID=IREAD(' Gi tidspunktet for kartet: ',29)
IF(ITID.LE.0) GOTO 20
TID=ITID
IMAX1=IRANGE/10+1
JMAX1=JRANGE/10+1
IMAX2=IRANGE/10
JMAX2=JRANGE/10
IHMIN=30000
IHMAX=0

C INNLESNING AV E-MATRIFSEFIL-----
EFILE(4)=48+IBAS
```

```

CALL OPEN(9,EFIL,0)
READ(9,24,ERR=98,END=99) IORD, IKURV
24 FORMAT(2I1)
IKURV1=IKURV
IF(IKURV.EQ.2) IKURV1=3
DO 21 I=1,IORD
  READ(9,22,ERR=98,END=99) (EMAT(I,K),K=1,IKURV1)
22 FORMAT(10E15.9)
  WRITE(3,23) (EMAT(I,K),K=1,IKURV1)
23 FORMAT(1X,10E15.9)
21 CONTINUE
ENDFILE 9
CALL LFEED(2)
C INNLESNING AV H-MATRISSEFIL-----
HFILE(5)=48+IBAS
CALL OPEN(7,HFILE,0)
READ(7,25,ERR=98,END=99) IHORD
25 FORMAT(I15)
IH=IHORD+1
DO 26 I=1,IH
  READ(7,27,ERR=98,END=99) (HMAT(I,J),J=1,IH)
  WRITE(3,28) (HMAT(I,J),J=1,IH)
27 FORMAT(6F15.9)
28 FORMAT(1X,6E15.9)
26 CONTINUE
ENDFILE 7
CALL LFEED(2)
C HØYDEBEREGNINGSLØSKE-----
DO 33 II=1,IMAX1
  IKOORD(II)=FORIGO+IOBS*(II-1)
33 CONTINUE
  WRITE(3,34) (IKOORD(II),II=1,IMAX1)
34 FORMAT(3X,19I4)
  DO 31 JJJ=1,JMAX1
  JJ=JMAX1-JJJ+1
  DO 32 II=1,IMAX1
    XKOORD=(IORIGO+10*(II-1))
    YKOORD=(JORIGO+10*(JJ-1)-6000)
    XKOORD=XKOORD/1000
    YKOORD=YKOORD/1000
    JKOORD=(YKOORD-JKOORD)*100
  IF(JKOORD.GE.100) JKOORD=JKOORD-100
  C BEREGNING AV STEDMATRISE FOR PUNKTET X,Y-----
  SVEKT(1)=1.
  SVEKT(2)=XKOORD
  SVEKT(3)=YKOORD
  SVEKT(4)=XKOORD*XKOORD
  SVEKT(5)=XKOORD*YKOORD
  SVEKT(6)=YKOORD*YKOORD
  SVEKT(7)=XKOORD*XKOORD*XKOORD
  SVEKT(8)=XKOORD*XKOORD*YKOORD
  SVEKT(9)=YKOORD*YKOORD*YKOORD
  SVEKT(10)=YKOORD*XKOORD*YKOORD
  C BEREGNING AV KOFFISIENTMATRISE FOR STEDET X,Y-----
  DO 41 I=1,IORD
    AAASUM=0.0
    DO 42 K=1,IKURV1
      BBB=EMAT(I,K)
      CCC=SVEKT(K)
      AAASUM=AAASUM+BBB*CCC
    42 CONTINUE
    DVEKT(I)=AAASUM
  41 CONTINUE
  C BEREGNING AV HOVEDFLATA FOR STEDET X Y-----
  ZZHL=ZHLCAL(XKOORD,YKOORD,IHORD,HMAT)
  IF(ZZHL.LT.0.0) ZZHL=0.0
  ZHLVAR=0.0
  C BEREGNING AV HAVNIVÅHØYDE PÅ STEDET X Y-----
  ZNO=HOYCAL(TID,IORD,DVEKT)
  ZNOVAR=0.0
  IF(ILOG.EQ.0)
    * CALL LINTRA(HOYD,HOYD1,HOYD2,ZZHL,ZHLVAR,ZNO,ZNOVAR)
  IF(ILOG.EQ.1)
    * CALL LOGTRA(HOYD,HOYD1,HOYD2,ZZHL,ZHLVAR,ZNO,ZNOVAR)
  IF(HOYD.LT.0.0) HOYD=0.0
  HAVET(II,JJ)=HOYD
  IHAVET(II,JJ)=HOYD
  IHMAX=MAX0(IHMAX,IHAVET(II,JJ))
  IHMIN=MIN0(IHMIN,IHAVET(II,JJ))
  32 CONTINUE
  WRITE(3,35) JKOORD,(IHAVET(K,JJ),K=1,IMAX1)
  35 FORMAT(1X,I2,19(1X,I3))
  31 CONTINUE
  C BEREGNING AV PAKKEDE VERDIER-----
  FKONST=(IHMAX-IHMIN)
  FKONST=255./FKONST
  WRITE(3,61) IHMIN
  61 FORMAT(/' Minste verdi: ',I10)
  WRITE(3,62) IHMAX
  62 FORMAT(' Største verdi: ',I10/)
  DO 63 I=1,IMAX1
  DO 63 J=1,JMAX1
    IHAVET(I,J)=FKONST*(HAVET(I,J)-IHMIN)
    IF(IHAVET(I,J).GT.255) IHAVET(I,J)=255
  63 CONTINUE
  C PLOTTING-----
  WRITE(3,71)
  71 FORMAT(' 0: Utplottning'/
    * ' 1: Nytt tidspunkt'/
    * ' 2: Bare skrivning av parameterfil'/)
  NEXT=IREAD(' ',4)
  IF(NEXT-1) 72,20,73
  72 CALL RUTMAL(IHAVET)
  73 CALL ENKODE
  CALL FCHAIN('MENYPRZCOM',0)
  C ERROR-----
  98 CALL TWRITE('***** READ-ERROR *****',25)
  STOP
  99 CALL TWRITE('***** END-OF-FILE *****',25)
  STOP
  END

```



```
C*****
C
C Program under PALEKART for innlesning av daterte MG-lokaliteter
C
C Programert av Kjell Kjenstad, Uio
C Januar 1983
C
C*****
PROGRAM ISOLLES
  BYTE NAVN(15),FILE(11)
  INTEGER IHAVET(19,13),IXX(300),IYY(300),ITID(300),IMIN(4),IH(19)
  REAL RMIN(4),XX,YY,ZZ,TT
  COMMON IPAGE,IBASE,IOBS,IHMAX,IHMIN,IX(7),JY(7)
  DATA FILE/'M','G',2*'Y',4*' ','D','A','T'/'
C SDI-----
  CALL GRAFIX
  IPAGE=1
  CALL SDINI
  CALL MAP4
C HEADING-----
  CALL TDV(25)
  WRITE(3,12)
  12 FORMAT(
    *,8X,'*****',
    *,8X,'**',
    *,8X,'**',
    *,8X,'**',
    *,8X,'**',
    *,8X,'*****',
    */)
    Program for innlesning av
    daterte MG-lokaliteter
    *****
C LESING AV PARAMETRE-----
  CALL DEKODE
  CALL PARAME
C PARAMETERBEREGNING-----
  IARUTE=IX(7)
  JARUTE=JY(7)
  IPRUTE=IX(6)
  JPRUTE=JY(6)
  IORIGO=IX(IARUTE+2)
  JORIGO=JY(JARUTE+2)
  IRANGE=((IX(IARUTE+3)-IX(IARUTE+2))*IOBS)/10
  JRANGE=((JY(JARUTE+3)-JY(JARUTE+2))*IOBS)/10
  IMAX1=IRANGE/10+1
  JMAX1=JRANGE/10+1
  IMAX2=IRANGE/10
  JMAX2=JRANGE/10
C INNLESNING AV DATA-----
  20 FILE(5)=48+IREAD(' Gi databasennummer: ',20)
  CALL OPEN(8,FILE,0)
  DO 21 I=1,300
    READ(8,22,END=23) III,(NAVN(K),K=1,15),XX,YY,ZZ,ZZHL
    *,TT,TTSD
  22 FORMAT(I3,1X,15A1,2X,F4.0,F5.0,1X,2(F4.0,1X),F6.0,2X,F4.0)
  IXX(I)=XX-1000.*IBASE
  IYY(I)=YY
  ITID(I)=TT/10
C *****
C
C Program under PALEKART for innlesning av daterte MG-lokaliteter
C
C Programert av Kjell Kjenstad, Uio
C Januar 1983
C
C*****
C BEREGNINGSLØKKE-----
  30 DO 31 JJJJ=1,JMAX1
    J=JMAX1-JJJJ+1
    JJ=JORIGO+10*(J-1)
    DO 32 I=1,IMAX1
      II=IORIGO+10*(I-1)
      DO 33 K=1,4
        RMIN(K)=IE8
        IMIN(K)=0
        CONTINUE
        IHAVET(I,J)=0
      33 CONTINUE
    C FINNER NÆRMESTE LOKALITET I HVER KVADRANT-----
      DO 34 K=1,IANT
        RID=IXX(K)-II
        RJD=IYY(K)-JJ
        IF((RID.GE.0.0).AND.(RJD.GE.0.0)) KVADR=1
        IF((RID.LT.0.0).AND.(RJD.GE.0.0)) KVADR=2
        IF((RID.LT.0.0).AND.(RJD.LT.0.0)) KVADR=3
        IF((RID.GE.0.0).AND.(RJD.LT.0.0)) KVADR=4
        R=(RID*RJD+RJD*RJD)
        IF(R.GT.RMIN(KVADR)) GOTO 34
        RMIN(KVADR)=R
        IMIN(KVADR)=K
      34 CONTINUE
    C FINNER GJENNOMSNITTSVERDIEN-----
      VEKT=0.0
      HAVET=0.0
      DO 35 K=1,4
        KHJ=IMIN(K)
        RHJ=SQRT(RMIN(K))
        IF(KHJ.EQ.0) GOTO 35
        IF(RHJ.EQ.0.0) RHJ=1.0
        HJELP=RHJ
        HJELP=1/HJELP
        VEKT=HJELP+VEKT
        HJELP=ITID(KHJ)
        HJELP=HJELP/RHJ
        HAVET=HJELP+HAVET
      35 CONTINUE
      HAVET=HAVET/VEKT
      IHAVET(I,J)=HAVET
      IH(I)=HAVET/10
    32 CONTINUE
  31 CONTINUE
  WRITE(3,36) JJJJ,IH(K),K=1,IMAX1
  36 FORMAT(1X,I2,1X,19I4)
C GLATTER FLATA-----
  DO 40 IIII=1,10
    CALL LFEED(1)
    IHMIN=30000
    IHMAX=-30000
    IIANT=0
  40 IIII=1,10
```

```

DO 41 JUUJ=1, JMAX1
J=JMAX1-JUUJ+1
DO 42 I=1, IMAX1
IF (I.EQ.1) GOTO 44
IF (I.EQ.IMAX1) GOTO 44
IF (J.EQ.1) GOTO 44
IF (J.EQ.JMAX1) GOTO 44
IMAXIM=0
IMINIM=26000
ISUM=0
DO 43 K=1, 3
DO 43 L=1, 3
  II=I+K-2
  JJ=J+L-2
  IMAXIM=MAX0(IMAXIM, IHAVET(II, JJ))
  IMINIM=MIN0(IMINIM, IHAVET(II, JJ))
  ISUM=ISUM+IHAVET(II, JJ)
CONTINUE
43 IF ((IHAVET(I, J).NE. IMAXIM).AND. (IHAVET(I, J).NE. IMINIM))
* GOTO 44
ISUM=ISUM/9
IIANT=IIANT+1
IHAVET(I, J)=ISUM
IH(I)=IHAVET(I, J)/10
IHMAX=MAX0(IHMAX, IHAVET(I, J))
IHMIN=MIN0(IHMIN, IHAVET(I, J))
42 CONTINUE
WRITE(3, 45) JUUJ, (IH(K), K=1, IMAX1)
45 FORMAT(1X, I2, 1X, 19I4)
41 CONTINUE
IF(IIANT.EQ.0) GOTO 60
40 CONTINUE
C BEREGNING AV PAKKEDE VERDIER-----
60 FKONST=(IHMAX-IHMIN)
FKONST=255./FKONST
WRITE(3, 67) IHMIN
67 FORMAT(// ' Minste verdi: ', I10//)
WRITE(3, 68) IHMAX
68 FORMAT(' Største verdi: ', I10//)
DO 63 I=1, IMAX1
DO 63 J=1, JMAX1
  IHAVET(I, J)=FKONST*(IHAVET(I, J)-IHMIN)
63 CONTINUE
C VALG-----
WRITE(3, 71)
71 FORMAT(' 0: Plott/'
* ' 1: Bare utskrift/'
* ' 2: Tilbake til hovedmeny'//)
NEXT=IREAD(' ', 7)
CALL TDV(25)
IF(NEXT-1) 72, 73, 90
72 CALL RUTMAL(IHAVET)
73 CALL ENKODE
C UTSKRIFT-----
90 CALL FCHAIN('MENYPRZCCOM', 0)
END

```

```

C*****
C Program under PALEKART for glatting av flate
C Programert av Kjell Kjenstad, Uio
C Januar 1983
C*****
PROGRAM RUTMID
INTEGER IHAVET(19,13)
COMMON IPAGE, IBASE, IOBS, IHMAX, IHMIN, IX(7), JY(7)
C SDI-----
CALL GRAFIX
IPAGE=1
CALL SDIINI
CALL MAP4
C HEADING-----
CALL TDV(25)
WRITE(3, 12)
12 FORMAT(
* 8X, '*****'
* 8X, '**'
* 8X, '**'
* 8X, '**'
* 8X, '**'
* 8X, '**'
* 8X, '*****'
*/)
Program for glatting av
flate
C LESING AV PARAMETRE-----
CALL DEKODE
CALL PARAME
C BEHANDLING AV PARAMETRE-----
IARUTE=IX(7)
JARUTE=JY(7)
IPRUTE=IX(6)
JPRUTE=JY(6)
IRANGE=(IX(IARUTE+3)-IX(IARUTE+2))*IOBS/10
JRANGE=(JY(JARUTE+3)-JY(JARUTE+2))*IOBS/10
IMAX2=IRANGE/IOBS
JMAX2=JRANGE/IOBS
C PARAMETERBEREGNING-----
CALL RUTLES(IHAVET)
DO 21 J=1, JMAX2
DO 21 I=1, IMAX2
  A=IHAVET(I, J)
  B=IHAVET(I+1, J)
  C=IHAVET(I, J+1)
  D=IHAVET(I+1, J+1)
  BB=(B-A)/10.
  CC=(C-A)/10.
  DD=(A+D-B-C)/100.
  IKONST=10*(I-1)
  JKONST=10*(J-1)
  DO 22 K=0, 9
  DO 22 L=0, 9
    IVERDI=(A+BB*K+CC*L+DD*L*K)
    II=IKONST+K+L

```

```
JJ=JKONST+L+1
CALL INPUT(II,JJ,1,IVERDI)
22 CONTINUE
21 CONTINUE
CALL FCHAIN('MENYPRZZCOM',0)
END
```

```
C*****
C Program under PALEKART for tegning av hav-paleokart
C
C Programmet av Kjell Kjenstad, Uio
C Januar 1983
C
C*****
PROGRAM HAVTOP
COMMON IPAGE,IEBASE,IOBS,IHMAX,IHMIN,IX(7),JY(7)
C SDI-----
CALL GRAFIX
IPAGE=0
CALL SDIINI
CALL MAP4
C HEADING-----
CALL TDV(25)
WRITE(3,12)
12 FORMAT(
*,8X,*****
*,8X,**,
*,8X,**,
*,8X,**,
*,8X,**,
*,8X,**,
*/)
Program for uttegning av
hav-paleokart
C*****
C LESING AV PARAMETRE-----
CALL DEKODE
CALL PARAME
C BEREGNING AV PARAMETRE-----
IARUTE=IX(7)
JARUTE=JY(7)
IPRUTE=IX(6)
JPRUTE=JY(6)
IORIGO=IX(IARUTE+2)
JORIGO=JY(IARUTE+2)
IRANGE=((IX(IARUTE+3)-IX(IARUTE+2))*IOBS)/10
JRANGE=((JY(JARUTE+3)-JY(JARUTE+2))*IOBS)/10
C EKVIDIANSSEDIALOG-----
CALL EKVIDI(IEKV,IEKVMI)
A=IEKV
A=10./A
B=255.*IEKV
B=(IHMAX-IHMIN)/B
C=IEKV
C=(-IHMIN+IEKV-IEKVMI-1)/C
C TEGNE-LØKKE-----
DO 21 J=1,JRANGE
DO 21 I=1,IRANGE
CALL WORKON(1)
IHAV=INTGET(I,J,1)
ITOP=INTGET(I,J,0)
CALL WORKON(0)
ITOP=A*ITOP-B*IHAV+C
IF(ITOP.LT.0) ITOP=0
IF(ITOP.GT.7) ITOP=7
```

```
CALL BYTPUT(I,J,I,PRUTE,J,PRUTE,I,TOP)
21 CONTINUE
C SLUTT-----
CALL FCHAIN('MENYPRZCOM',0)
END

C*****
C Program under PALEKART for tegning av isobaskart
C
C Programert av Kjell Kjenstad, Uio
C November 1983
C
C*****
PROGRAM ISOBAS
BYTE IARRAY(700)
COMMON IPAGE,IBASE,IOBS,IHMAX,IHMIN,IX(7),JY(7)
C SDI-----
CALL GRAFIX
IPAGE=0
CALL SDIINI
CALL MAP4
C HEADING-----
CALL TDV(25)
10 WRITE(3,11)
11 FORMAT(
* 8X,'*****',
* 8X,'**',
* 8X,'**',
* 8X,'**',
* 8X,'**',
* 8X,'**',
*)
Program for tegning av
isobaskart
C LESING AV PARAMETRE-----
CALL DEKODE
CALL PARAME
C BEREGNING AV PARAMETRE-----
IARUTE=IX(7)
JARUTE=JY(7)
IPRUTE=IX(6)
JPRUTE=JY(6)
IORIGO=IX(IARUTE+2)
JORIGO=JY(IARUTE+2)
IRANGE=((IX(IARUTE+3)-IX(IARUTE+2))*IOBS)/10
JRANGE=((JY(JARUTE+3)-JY(JARUTE+2))*IOBS)/10
C EKVIDISTANSEDIALOG-----
CALL EKVIDI(IEKV,IMINI)
IEKV=IEKV/10
IMINI=IMINI/10
IKONST=IEKV-IMINI-1
C OMRGNINGSKONSTANTER-----
C=(IHMAX-IHMIN)
C=C/255.
D=IHMIN
C UTVISKINGSLØKKE-----
IJA=IREAD('Tegning av nytt kart: (Ja)',27)
IF(IJA.EQ.0) GOTO 20
DO 12 J=1,JRANGE
DO 12 I=1,IRANGE
CALL WORKON(1)
ITOP=INTGET(I,J,0)
ITOP=(ITOP+IKONST)/IEKV
```

```

IF (ITOP.LT.0) ITOP=0
IF (ITOP.GT.7) ITOP=7
CALL WORKON(0)
CALL BYTPUT(I,J,IPRUTE,JPRUTE,ITOP)
12 CONTINUE
C ISOBASHØYDE-----
20 ISOBAS=IREAD(' Gi isobashøyde:',16)
IF(ISOBAS.EQ.0) GOTO 99
C INITIALISERING AV LETEPARAMETRE-----
IRETN=4
INEXT=1
JNEXT=1
IPUNKT=1
C OPPBYGGING AV FRONTARRAY-----
30 CALL WORKON(1)
IMI=IRETN-3
IMA=IRETN+3
DO 31 I=IMI,IMA
  II=I
  CALL NESTE(INEXT,JNEXT,INY,JNY,II)
  IHAV=INTGET(INY,JNY,I)
  HAVNY=C*IHAV+D
  IF(I.EQ.IMI) GOTO 34
  IF((INY.LE.1).OR.(JNY.GE.JRANGE)).OR.
    ((JNY.LE.1).OR.(JNY.GE.JRANGE)) GOTO 32
  IF((HAVGL.LE.ISOBAS).AND.(HAVNY.GT.ISOBAS)) GOTO 32
  IF((HAVGL.GT.ISOBAS).AND.(HAVNY.LE.ISOBAS)) GOTO 32
34 HAVGL=HAVNY
31 CONTINUE
32 IRETN=II
IF(HAVGL.LE.ISOBAS) IRETN=II-1
IF((INY.LE.1).OR.(JNY.GE.JRANGE)).OR.
  ((JNY.LE.1).OR.(JNY.GE.JRANGE)) IRETN=II
CALL WORKON(0)
CALL NESTE(INEXT,JNEXT,INY,JNY,IRETN)
IARRAY(IPUNKT)=IRETN
IF(IPUNKT.GE.700) GOTO 40
IF((INY.EQ.2).AND.(JNY.EQ.1)) GOTO 40
IF((INY.NE.1).AND.(JNY.NE.JRANGE)).AND.
  ((JNY.NE.1).AND.(JNY.NE.JRANGE)) GOTO 35
IF((INEXT.NE.1).AND.(INEXT.NE.IRANGE)).AND.
  ((JNEXT.NE.1).AND.(JNEXT.NE.JRANGE)) GOTO 40
35 IF((INY.NE.1).AND.(INY.NE.IRANGE)).AND.
  ((JNY.NE.1).AND.(JNY.NE.JRANGE))
  * CALL BYTPUT(INY,JNY,IPRUTE,JPRUTE,15)
INEXT=INY
JNEXT=JNY
IPUNKT=IPUNKT+1
GOTO 30
C UTVIKING AV ISOBAS-----
40 KMAX=IPUNKT
IEND=INY
JEND=JNY
ISTEP=0
IJA=LREAD(' OK isobas: (Ja)',17)
IF(IJA.NE.0) GOTO 20

```

```

INEXT=1
JNEXT=1
DO 41 I=1,KMAX
  ITEMP=IARRAY(I)
  CALL NESTE(INEXT,JNEXT,INY,JNY,ITEMP)
  INEXT=INY
  JNEXT=JNY
  IF((INY.EQ.2).AND.(JNY.EQ.1)) GOTO 20
  CALL WORKON(1)
  ITOP=INTGET(INY,JNY,0)
  ITEMP=(ITOP+IKONST)/IEKV
  IF(ITOP.LT.0) ITOP=0
  IF(ITOP.GT.7) ITOP=7
  CALL WORKON(0)
  CALL BYTPUT(INY,JNY,IPRUTE,JPRUTE,ITOP)
41 CONTINUE
GOTO 20
C SLUTT-----
99 CALL FCHAIN('MENYPRZCOM',0)
END
C SUBROUTINE FOR BEREGNING AV NESTE KOORDINAT-----
SUBROUTINE NESTE(IGL,JGL,INY,JNY,IRETN)
IF(IRETN.LT.0) IRETN=IRETN+8
IF(IRETN.GT.7) IRETN=IRETN-8
ITEMP=IRETN+1
GOTO (10,20,30,40,50,60,70,80),ITEMP
10 INY=IGL+1
JNY=JGL
RETURN
20 INY=IGL+1
JNY=JGL+1
RETURN
30 INY=IGL
JNY=JGL+1
RETURN
40 INY=IGL-1
JNY=JGL+1
RETURN
50 INY=IGL-1
JNY=JGL
RETURN
60 INY=IGL-1
JNY=JGL-1
RETURN
70 INY=IGL
JNY=JGL-1
RETURN
80 INY=IGL+1
JNY=JGL-1
RETURN
END

```

```
C*****
C
C Program under PALEKART for uttegning av kart over marine områder
C
C Programmet av Kjell Kjenstad, Uio
C Januar 1983
C
C*****
PROGRAM MARINE
COMMON IPAGE,IBASE,IOBS,IHMAX,IHMIN,IX(7),JY(7)
C SDI-----
CALL GRAFIX
IPAGE=0
CALL SDINI
CALL MAP4
C HEADING-----
CALL TDV(25)
WRITE(3,12)
12 FORMAT(
*,8X,'*****'
*,8X,'**'
*,8X,'**'
*,8X,'**'
*,8X,'**'
*,8X,'**'
*,8X,'*****'
*)
C LESING AV PARAMETRE-----
CALL DEKODE
CALL PARAME
C BEREGNING AV PARAMETRENE-----
IARUTE=IX(7)
JARUTE=JY(7)
IPRUTE=IX(6)
JPRUTE=JY(6)
IORIGO=IX(IARUTE+2)
JORIGO=JY(JARUTE+2)
IRANGE=((IX(IARUTE+3)-IX(IARUTE+2))*IOBS)/10
JRANGE=((JY(JARUTE+3)-JY(JARUTE+2))*IOBS)/10
C TOPOGRAFIDIALOG-----
CALL TWRITE(' Topografialog:',17)
CALL LFEED(1)
CALL EKVIDI(IEKV,IEKVM1)
IA=IEKV-IEKVM1-1
B=IEKV
B=1/B
C=(IHMAX-IHMIN)
C=C/255.
C PLOTTELØKKE-----
DO 21 J=1,JRANGE
DO 21 I=1,IRANGE
CALL WORKON(1)
ITOP=INTGET(I,J,0)
ITOP=ITOP*10
IHAV=INTGET(I,J,1)
IHAV=C*IHAV+IHMIN
ITOP=ITOP+IA
IMAR=ITOP*B
CALL WORKON(0)
IF(IMAR.LE.0) IMAR=0
IF(IMAR.GE.7) IMAR=7
CALL BYTPUT(I,J,IPRUTE,JPRUTE,IMAR)
21 CONTINUE
C SLUTT-----
CALL FCHAIN('MENYPRZZCOM',0)
END
```

```

C*****
C
C Program under PALEKART for transformasjon til paleo-topobase
C
C Programmet av Kjell Kjenstad, UiO
C Januar 1983
C
C*****
PROGRAM KARPAL
BYTE HFILE(11), EFILE(11)
REAL EMAT(7,10), SVEKT(10), DVEKT(7), HMAT(5,5), HAVET(19,13)
DOUBLE PRECISION AAASUM, AAHOY, BBB, CCC, DDD
INTEGER IHAVET(19,13), IKOORD(18)
COMMON IPAGE, IBASE, IOBS, IHMAX, IHMIN, IX(7), JY(7)
DATA EFILE/'E', 2*'X', 5*' ', 'M', 'A', 'T'/'
DATA HFILE/'H', 'L', 2*'X', 4*' ', 'M', 'A', 'T'/'
C SDI-----
CALL GRAFIX
IPAGE=1
CALL SDIINI
CALL MAP4
C HEADING-----
CALL TDV(25)
WRITE(3,12)
12 FORMAT(
*, 8X, '*****'
*, 8X, '***'
*, 8X, '***'
*, 8X, '***'
*, 8X, '*****'
*)
C LESING AV PARAMETRE-----
CALL DEKODE
CALL PARAME
C PARAMETERBEREGNING-----
IARUTE=IX(7)
JARUTE=JY(7)
IPRUTE=IX(6)
JPRUTE=JY(6)
IORIGO=IX(IARUTE+2)
JORIGO=JY(JARUTE+2)
IRANGE=((IX(IARUTE+3)-IX(IARUTE+2))*IOBS)/10
JRANGE=((JY(JARUTE+3)-JY(JARUTE+2))*IOBS)/10
C PARAMETERDIALOG-----
20 IBAS=IREAD(' Gi databasenummer: ', 20)
ILOG=IREAD(' Logaritmetransformasjon (Ja) ', 29)
ITID=IREAD(' Gi tidspunktet for kartet: ', 29)
IF(ITID.LE.0) GOTO 20
TID=ITID
IMAX1=IRANGE/10+1
JMAX1=JRANGE/10+1
IMAX2=IRANGE/10
JMAX2=JRANGE/10
IHMIN=30000
IHMAX=0
C *****
C INNLESNING AV E-MATRISSEFIL-----
EFILE(4)=48+IBAS
CALL OPEN(9,EFILE,0)
READ(9,24,ERR=98,END=99) IORD, IKURV
IKURV1=IKURV
IF(IKURV.EQ.2) IKURV1=3
DO 21 I=1,IORD
READ(9,22,ERR=98,END=99) (EMAT(I,K), K=1, IKURV1)
22 FORMAT(10E15.9)
WRITE(3,23) (EMAT(I,K), K=1, IKURV1)
23 FORMAT(1X,10E15.9)
21 CONTINUE
ENDFILE 9
CALL LFEED(2)
C INNLESNING AV H-MATRISSEFIL-----
HFILE(5)=48+IBAS
CALL OPEN(7,HFILE,0)
READ(7,25,ERR=98,END=99) IHORD
25 FORMAT(I15)
IH=IHORD+1
DO 26 I=1,IH
READ(7,27,ERR=98,END=99) (HMAT(I,J), J=1, IH)
WRITE(3,28) (HMAT(I,J), J=1, IH)
27 FORMAT(6F15.9)
28 FORMAT(1X,6E15.9)
26 CONTINUE
ENDFILE 7
CALL LFEED(2)
C HØYDEBEREGNINGSLØSKE-----
DO 33 II=1,IMAX1
IKOORD(II)=IORIGO+IOBS*(II-1)
33 CONTINUE
WRITE(3,34) (IKOORD(II), II=1, IMAX1)
34 FORMAT(3X,19I4)
DO 31 JJJ=1, JMAX1
JJ=JMAX1-JJJ+1
DO 32 II=1, IMAX1
XKOORD=(IORIGO+10*(II-1))
YKOORD=(JORIGO+10*(JJ-1)-6000)
XKOORD=XKOORD/1000
YKOORD=YKOORD/1000
JKOORD=(YKOORD-JKOORD)*100
IF(JKOORD.GE.100) JKOORD=JKOORD-100
C BEREGNING AV STEDMATRISE FOR PUNKTET X, Y-----
SVEKT(1)=1.
SVEKT(2)=XKOORD
SVEKT(3)=YKOORD
SVEKT(4)=XKOORD*XKOORD
SVEKT(5)=XKOORD*YKOORD
SVEKT(6)=YKOORD*YKOORD
SVEKT(7)=XKOORD*XKOORD*XKOORD
SVEKT(8)=XKOORD*YKOORD*YKOORD
SVEKT(9)=YKOORD*YKOORD*YKOORD
SVEKT(10)=YKOORD*XKOORD*YKOORD

```

```

C BEREGNING AV KOFFISIENTMATRISE FOR STEDET X,Y-----
DO 41 I=1,IORD
  AAASUM=0.0
  DO 42 K=1,IKURV1
    BBB=EMAT(I,K)
    CCC=SVEKT(K)
    AAAASUM=AAAASUM+BBB*CCC
  42 CONTINUE
  DVEKT(I)=AAAASUM
  41 CONTINUE
C BEREGNING AV HOVEDFLATA FOR STEDET X Y -----
  ZZHL=ZHLCAL(XKOORD,YKOORD,IHORD,HMAT)
  IF(ZZHL.LT.0.0) ZZHL=0.0
  ZHLVAR=0.0
C BEREGNING AV HAVNIVÅHØYDE PÅ STEDET X Y-----
  ZNO=HOYCAL(TID,IORD,DVEKT)
  ZNOVAR=0.0
  IF(ILOG.EQ.0)
    * CALL LINTRA(HOYD,HOYD1,HOYD2,ZZHL,ZHLVAR,ZNO,ZNOVAR)
  IF(ILOG.EQ.1)
    * CALL LOGTRA(HOYD,HOYD1,HOYD2,ZZHL,ZHLVAR,ZNO,ZNOVAR)
  IF(HOYD.LT.0.0) HOYD=0.0
  HAVET(II,JJ)=HOYD
  IHAVET(II,JJ)=HOYD
  IHMAX=MAX0(IHMAX,IHAVET(II,JJ))
  IHMIN=MIN0(IHMIN,IHAVET(II,JJ))
  32 CONTINUE
  WRITE(3,35) JKOORD,(IHAVET(K,JJ),K=1,IMAX1)
  35 FORMAT(1X,I2,19(1X,I3))
  31 CONTINUE
C BEREGNINGSLØKKE-----
DO 51 J=1,JRANGE
DO 51 I=1,IRANGE
  II=I/10+1
  JJ=J/10+1
  ITOP=INTGET(I,J,0)
  IHAV=IHAVET(II,JJ)
  IHAV=IHAV/10
  ITOP=ITOP-IHAV
  IF(ITOP.LE.0) ITOP=0
  IF(ITOP.GT.255) ITOP=255
  CALL INTPUT(I,J,0,ITOP)
  51 CONTINUE
C SLUTT-----
C ERROR-----
  98 CALL TWRITE('***** READ-ERROR *****',25)
  STOP
  99 CALL TWRITE('***** END-OF-FILE *****',25)
  STOP
  END
C*****
C Program under PALEKART for beregning av iskropp
C Programmet av Kjell Kjenstad, Uio
C Januar 1983
C*****
PROGRAM ISFLAT
INTEGER ITID(19,13),ISNY(19,13),ISGL(19,13),IT(19)
COMMON IPAGE,IBASE,IOBS,IHMAX,IHMIN,IX(7),JY(7)
C SDI-----
CALL GRAFIX
IPAGE=1
CALL SDIINI
  10 CALL MAP4
C HEADING-----
CALL TDV(25)
WRITE(3,11)
  11 FORMAT(
    * ,8X,'*****
    * ,8X,'**
    * ,8X,'**
    * ,8X,'**
    * ,8X,'**
    * ,8X,'*****
    *)
    Program for beregning av
    iskropp
C LESING AV PARAMETRE-----
CALL DEKODE
CALL PARAME
C BEREGNING AV PARAMETRE-----
  IARUTE=IX(7)
  JARUTE=JY(7)
  IPRUTE=IX(6)
  JPRUTE=JY(6)
  IRANGE=((IX(IARUTE+3)-IX(IARUTE+2))*IOBS)/10
  JRANGE=((JY(JARUTE+3)-JY(JARUTE+2))*IOBS)/10
  IMAX2=IRANGE/IOBS+1
  JMAX2=JRANGE/IOBS+1
C DIALOG-----
  IFRONT=IREAD(' Gi tidspunkt for kartet: ',26)
  IFRONT=IFRONT/10
  ISTEP=IREAD(' Gi høyde på 10 km-step (450): ',31)
  IF(ISTEP.LT.0) GOTO 10
  IF(ISTEP.EQ.0) ISTEP=450
  ISTEP=ISTEP/10
C BEREGNING AV TID-----
CALL RUTLES(ITID)
FKONST=IHMAX-IHMIN
FKONST=FKONST/255.
CALL LFEED(2)
CALL TWRITE(' Tidfløte: ',10)
CALL LFEED(1)
DO 13 JJ=1,JMAX2
  J=JMAX2-JJ+1
DO 14 I=1,IMAX2

```



```

ITID(I,J)=ITID(I,J)*FKONST+IHMIN
IT(I)=ITID(I,J)/10
14 CONTINUE
WRITE(3,15) JJ,(IT(K),K=1,IMAX2)
15 FORMAT(1X,I2,1X,19I4)
13 CONTINUE
C IS/IKKE-IS
CALL LFEED(2)
CALL TWRITE(' Isfrontflate:',14)
CALL LFEED(1)
DO 21 JJ=1,JMAX2
J=JMAX2-JJ+1
DO 22 I=1,IMAX2
ISGL(I,J)=0
IF(ITID(I,J).GT.IFRONT) ISGL(I,J)=-ISTEP
22 CONTINUE
WRITE(3,23) J,(ISGL(K,J),K=1,IMAX2)
23 FORMAT(1X,I2,1X,19I4)
21 CONTINUE
C OPPBYGGINGSLØKKE
LOOP=0
30 IANT=0
LOOP=LOOP+1
WRITE(3,29) LOOP
29 FORMAT(//1X,I2,'. oppbyggingsløkke: '//)
DO 31 I=1,IMAX2
DO 31 J=1,JMAX2
C-----VALG PÅ GRUNNLAG AV GAMMEL VERDI-----
IMAX=0
IMIN=0
IF(ISGL(I,J)) 53,32,53
C-----
32 IP1=I+1
JP1=J+1
IM1=I-1
JM1=J-1
IF(IM1.LT.1) IM1=1
IF(JM1.LT.1) JM1=1
IF(IP1.GT.IMAX2) IP1=IMAX2
IF(JP1.GT.JMAX2) JP1=JMAX2
IMAX=MAX0(ISGL(IM1,J),ISGL(IP1,J),ISGL(I,JM1),ISGL(I,JP1))
IMIN=MIN0(ISGL(IM1,J),ISGL(IP1,J),ISGL(I,JM1),ISGL(I,JP1))
IF(IMIN.LT.0) GOTO 51
IF(IMAX-1) 53,52,41
C-----NY VERDI-----
41 IANT=IANT+1
HJELP=IMAX
HJELP=HJELP*HJELP+2250
HJELP=SQRT(HJELP)
ISNY(I,J)=HJELP
GOTO 31
C-----FRONTVERDI-----
51 IANT=IANT+1
ISNY(I,J)=1
GOTO 31

```

```

C-----FØRSTE LØKKE-----
52 IANT=IANT+1
ISNY(I,J)=ISTEP
GOTO 31
C-----SAMME VERDI-----
53 ISNY(I,J)=ISGL(I,J)
31 CONTINUE
C OVERSKRIVING OG UTSKRIFT-----
60 IHMIN=30000
IHMAX=-30000
DO 61 JJ=1,JMAX2
J=JMAX2-JJ+1
DO 62 I=1,IMAX2
ISGL(I,J)=ISNY(I,J)
IHMAX=MAX0(IHMAX,ISGL(I,J))
IHMIN=MIN0(IHMIN,ISGL(I,J))
62 CONTINUE
WRITE(3,63) JJ,(ISGL(K,J),K=1,IMAX2)
63 FORMAT(1X,I2,1X,19I4)
61 CONTINUE
IF(IANT.EQ.0) GOTO 80
GOTO 30
C VALG-----
80 FKONST=255./(IHMAX-IHMIN)
WRITE(3,81) IHMIN
81 FORMAT(//' Minste verdi: ',I10/)
WRITE(3,82) IHMAX
82 FORMAT(' Største verdi: ',I10//)
WRITE(3,83)
83 FORMAT(' 0: plotting' /
* ' 1: ny flate' /
* ' 2: til hovedmeny')
NEXT=IREAD(' ',3)
IF(NEXT-1) 90,10,100
C PLOTTING PÅ SKJERM-----
90 DO 91 I=1,IMAX2
DO 91 J=1,JMAX2
ISGL(I,J)=FKONST*(ISGL(I,J)-IHMIN)
91 CONTINUE
CALL ENKODE
CALL RUTMAL(ISGL)
C SLUTT-----
100 CALL FCHAIN('ISFWDZCOM',0)
END

```

```

C*****
C
C Program under PALEKART for glatting av isflate
C
C Programert av Kjell Kjenstad, Uio
C Januar 1983
C
C*****
PROGRAM ISFMID
INTEGER IHAVET(19,13)
COMMON IPAGE,IBASE,IOBS,IHMAX,IHMIN,IX(7),JY(7)
C SDI-----
CALL GRAFIX
IPAGE=1
CALL SDIINI
CALL MAP4
C HEADING-----
CALL TDV(25)
WRITE(3,12)
12 FORMAT(
*,8X,'*****'
*,8X,'**'
*,8X,'**'
*,8X,'**'
*,8X,'**'
*,8X,'**'
*,8X,'*****'
*)
C LESING AV PARAMETRE-----
CALL DEKODE
CALL PARAME
C BEHANDLING AV PARAMETRE-----
IARUTE=IX(7)
JARUTE=JY(7)
IPRUTE=IX(6)
JPRUTE=JY(6)
IRANGE=((IX(IARUTE+3)-IX(IARUTE+2))*IOBS)/10
JRANGE=((JY(JARUTE+3)-JY(JARUTE+2))*IOBS)/10
IMAX1=IRANGE/IOBS+1
JMAX1=JRANGE/IOBS+1
IMAX2=IRANGE/IOBS
JMAX2=JRANGE/IOBS
C INNLESNING AV FLATE-----
CALL RUTLES(IHAVET)
FKONST=IHMAX-IHMIN
FKONST=FKONST/255.
DO 13 JJ=1,JMAX1
J=JMAX1-JJ+1
DO 14 I=1,IMAX1
IHAVET(I,J)=IHAVET(I,J)*FKONST+IHMIN
14 CONTINUE
WRITE(3,15) JJ,(IHAVET(K,J),K=1,IMAX1)
15 FORMAT(1X,I2,1X,19I4)
13 CONTINUE
CALL LFEED(1)
C GLATTINGSLØKKE-----
30 DO 31 JJ=1,JMAX1

```

```

J=JMAX1-JJ+1
DO 32 I=1,IMAX1
IP1=I+1
IM1=I-1
JP1=J+1
JM1=J-1
IF(IP1.GT.IMAX1) IP1=IMAX1
IF(JP1.GT.JMAX1) JP1=JMAX1
IF(IM1.LT.1) IM1=1
IF(JM1.LT.1) JM1=1
ISUM=IHAVET(IP1,J)+IHAVET(I,JP1)+IHAVET(IM1,J)+IHAVET(I,JM1)
HJELP=ISUM
HJELP=HJELP/4
IHAVET(I,J)=HJELP
32 CONTINUE
WRITE(3,15) JJ,(IHAVET(K,J),K=1,IMAX1)
31 CONTINUE
CALL LFEED(2)
C VALG-----
NEXT=IREAD(' 0: finginglatting 1:ny grovglatting 2: hovedmeny
*:',52)
IF(NEXT-1) 40,30,100
C BEREGNING AV PAKKEDE VERDIER-----
40 FKONST=(IHMAX-IHMIN)
FKONST=255./FKONST
DO 43 I=1,IMAX1
DO 43 J=1,JMAX1
IHAVET(I,J)=FKONST*(IHAVET(I,J)-IHMIN)
43 CONTINUE
C FINGLATTING-----
DO 51 J=1,JMAX2
DO 51 I=1,IMAX2
A=IHAVET(I,J)
B=IHAVET(I+1,J)
C=IHAVET(I,J+1)
D=IHAVET(I+1,J+1)
BB=(B-A)/10.
CC=(C-A)/10.
DD=(A+D-B-C)/100.
IKONST=10*(I-1)
JKONST=10*(J-1)
DO 52 K=0,9
DO 52 L=0,9
IVERDI=(A+BB*K+CC*L+DD*L*K)
II=IKONST+K+1
JJ=JKONST+L+1
CALL INTPUT(II,JJ,1,IVERDI)
52 CONTINUE
51 CONTINUE
C SLUTT-----
100 CALL FCHAIN(' ISKARTZZCOM',0)
END

```

```
C*****
C Program under PALEKART for tegning av is-paleokart
C
C Programmet av Kjell Kjenstad, UiO
C Januar 1983
C
C*****
PROGRAM ISKART
COMMON IPAGE,IBASE,IOBS,IHMAX,IHMIN,IX(7),JY(7)
C SDI-----
CALL GRAFIX
IPAGE=0
CALL SDINI
CALL MAP4
C HEADING-----
CALL TDV(25)
WRITE(3,12)
12 FORMAT(
*,8X,'*****',
*,8X,'**',
*,8X,'**',
*,8X,'**',
*,8X,'**',
*,8X,'**',
*,8X,'*****',
*)
PROGRAM for tegning av
is-paleokart
C LESING AV PARAMETRE-----
CALL DEKODE
CALL PARAME
C PARAMETERBEREGNING-----
IARUTE=IX(7)
JARUTE=JY(7)
IPRUTE=IX(6)
JPRUTE=JY(6)
IORIGO=IX(IARUTE+2)
JORIGO=JY(IARUTE+2)
IRANGE=((IX(IARUTE+3)-IX(IARUTE+2))*IOBS)/10
JRANGE=((JY(JARUTE+3)-JY(JARUTE+2))*IOBS)/10
FKONST=(IHMAX-IHMIN)
FKONST=FKONST/255.
C EKVIDISTANSEDIALOG-----
IEKV=IEKV/10
IMINI=IMINI/10
IKONST=IEKV-IMINI-1
DO 21 J=1,IRANGE
DO 21 I=1,IRANGE
CALL WORKON(1)
ITOP=INTGET(I,J,0)
ISTO=INTGET(I,J,1)
ISTO=ISTO*FKONST+IHMIN
IF(ISTO.LE.ITOP) IVAL=(ITOP+IKONST)/IEKV
IF(IVAL.GT.7) IVAL=7
CALL WORKON(0)
IF(ISTO.GT.ITOP) IVAL=(ISTO+IKONST)/IEKV+8
IF(IVAL.LT.0) IVAL=0
IF(IVAL.GT.15) IVAL=15
CALL BYTPUT(I,J,IPRUTE,JPRUTE,IVAL)
21 CONTINUE
C SLUTT-----
CALL FCHAIN('MENYPRZZCOM',0)
END
```

```

*****
**
**      Subrutinebibliotek
**      for FORTRAN-subrutiner
**
*****
SUBROUTINE TDV foretar sending av
      kodemeldinger for terminalhåndtering.....s. A - 52

SUBROUTINE LFEED foretar "line-feed" i
      terminal-dialoger.....s. A - 52

FUNCTION IREAD foretar innlesing av
      heltall under dialog.....s. A - 52

FUNCTION FREAD foretar innlesing av
      reelt tall under dialog.....s. A - 52

FUNCTION AREAD foretar innlesing av
      bokstav under dialog.....s. A - 53

FUNCTION LREAD foretar ja/nei-svar under dialog.....s. A - 53

SUBROUTINE FIREAD foretar innlesing av
      fil-navn under dialog.....s. A - 53

SUBROUTINE LEAST1 foretar minste kvadrats
      tilpasning av krum regressjon.....s. A - 53

SUBROUTINE INVERS foretar matriseinvertering
      ved LU-dekomposisjon.....s. A - 54

FUNCTION TIDCAL foretar tidspunktbestemming i
      den enkle havnivåmodellen.....s. A - 56

FUNCTION VARCAL foretar beregning av standard-
      avvik i den enkle havnivåmodellen.....s. A - 56

FUNCTION HOYCAL foretar høydeberegning i
      den enkle havnivåmodellen.....s. A - 56

FUNCTION ZHLCAL foretar beregning av trend-
      flatehøyde i den enkle havnivåmodellen.....s. A - 57

SUBROUTINE LINTRA foretar lineærtransformasjon av
      normaliserte strandforsyvningskurver.....s. A - 57

SUBROUTINE LOGTRA foretar logaritmetransformasjon av
      normaliserte strandforsyvningskurver.....s. A - 57

?????

```

```

C*****
C**
C** Subrutinebibliotek
C** for FORTRAN-subrutiner
C**
C** Programmert av Kjell Kjenstad
C** Desember 1982
C**
C*****
C
C
C*****
C** LINEFEED
C**
C*****
      SUBROUTINE LFEED(I)
      DO 20 J=1,I
      WRITE(3,30)
      30 FORMAT(
      20 CONTINUE
      RETURN
      END
C*****
C** LESNING AV INTEGRERTALL
C**
C*****
      FUNCTION IREAD(NAVN,I)
      BYTE NAVN(I)
      WRITE(3,30) (NAVN(K),K=1,I)
      30 FORMAT(60A1)
      READ(3,40) IREAD
      40 FORMAT(I15)
      RETURN
      END
C*****
C** LESNING AV REALTALL
C**
C*****
      FUNCTION FREAD(NAVN,I)
      BYTE NAVN(I)
      WRITE(3,30) (NAVN(K),K=1,I)
      30 FORMAT(60A1)
      READ(3,40) FREAD
      40 FORMAT(F15.7)
      RETURN
      END
C*****
C** LESNING AV ASCIITEGN
C**
C*****
      FUNCTION AREAD(NAVN,I)
      BYTE NAVN(I),AREAD

```

```

WRITE(3,30) (NAVN(K),K=1,I)
30 FORMAT(60A1)
READ(3,40) AREAD
40 FORMAT(1A1)
RETURN
END
C*****
C**
C** JA/NEI-SVAR
C**
C*****
FUNCTION LREAD(TEXT,N)
BYTE TEXT(N),NEXT
WRITE(3,21) (TEXT(K),K=1,N)
21 FORMAT(80A1)
READ(3,21) NEXT
LREAD=1
IF((NEXT.EQ.78).OR.(NEXT.EQ.110)) LREAD=0
RETURN
END
C*****
C**
C** INNESNING AV FIL-NAVN
C**
C*****
SUBROUTINE FIREAD(NAVN,TEXT)
BYTE NAVN(11),TEMP(12),TEXT(12)
WRITE(3,11) (TEXT(K),K=1,12)
11 FORMAT(' Gi fil-navn (' ,12A1, ' ) : ' )
READ(3,12) (TEMP(K),K=1,12)
22 FORMAT(12A1)
DO 19 I=1,12
19 CONTINUE
IF(TEMP(I).GT.96) TEMP(I)=TEMP(I)-32
DO 13 I=1,11
NAVN(I)=' '
13 CONTINUE
DO 14 I=1,9
INR=I
IF(TEMP(INR).EQ.46) GOTO 15
14 CONTINUE
15 IMJ=INR-1
NAVN(9)=TEMP(INR+1)
NAVN(10)=TEMP(INR+2)
NAVN(11)=TEMP(INR+3)
DO 16 J=1,IMJ
NAVN(J)=TEMP(J)
16 CONTINUE
WRITE(3,17) (NAVN(K),K=1,11)
17 FORMAT(' -->working: file opened: '8A1,'3A1)
RETURN
END
C*****
C**
C** MINSTE KVADRATS MATRISEINVERTERING
C**
C*****
SUBROUTINE LEAST1(TVEKT,IA,IB,N,M,YVEKT,XVEKT,COVARR,Q,BARR,IVEKT
*,ZVEKT,IERR)
REAL TVEKT(IA),YVEKT(IA),XVEKT(IB),ZVEKT(IB),BARR(IB,IB)
*,COVARR(IB,IB)
DOUBLE PRECISION AAASUM,QQQ,AAA,BBB,CCC,DDD,EEE,EENORM
INTEGER IVEKT(IB)
CALL TWRITE(' -->working: least square: begin ' ,36)
C INITIALISERING-----
DO 11 I=1,IB
XVEKT(I)=0.0
ZVEKT(I)=0.0
IVEKT(I)=0
DO 11 J=1,IB
BARR(I,J)=0.0
COVARR(I,J)=0.0
11 CONTINUE
C At*A-----
DO 21 I=1,M
DO 21 J=1,M
AAASUM=0.0
IEKSP=I+J
DO 22 K=1,N
BBB=TVEKT(K)**IEKSP
AAASUM=AAASUM+BBB
22 CONTINUE
BARR(I,J)=AAASUM
21 CONTINUE
C (At*A)-1-----
CALL TWRITE(' -->working: least square: inversion',36)
CALL INVERS(BARR,IB,M,COVARR,ZVEKT,IVEKT,IERR)
CALL TWRITE(' -->working: least square: inversion completed',46)
C Y*Y-----
AAASUM=0.0
DO 31 I=1,N
BBB=YVEKT(I)
AAASUM=AAASUM+BBB*BBB
31 CONTINUE
YHU=AAASUM
DO 41 I=1,M
AAASUM=0.0
DO 42 K=1,N
BBB=TVEKT(K)**I
CCC=YVEKT(K)
AAASUM=AAASUM+BBB*CCC
42 CONTINUE
ZVEKT(I)=AAASUM
41 CONTINUE
C ((At*A)-1)*(At*Y)-----
DO 51 I=1,M
AAASUM=0.0

```

```

DO 52 K=1,M
  BBB=COVARR(I,K)
  CCC=ZVEKT(K)
  AAASUM=AAASUM+BBB*CCC
52 CONTINUE
  XVEKT(I)=AAASUM
51 CONTINUE
  CALL TWRITE(' -->working: least square: x-vector completed ',46)
  ITERATIV FORBEDRING AV XVEKT-----
DO 61 I=1,M
  DO 61 J=1,M
  AAASUM=0.0
  IEKSP=I+J
  DO 62 K=1,N
  BBB=I*VEKT(K)**IEKSP
  AAASUM=AAASUM+BBB
62 CONTINUE
  BARR(I,J)=AAASUM
61 CONTINUE
  IANT=0
63 EENORM=0.0
  DO 64 I=1,M
  AAASUM=0.0
  DO 65 J=1,M
  QQQ=0.0
  DO 66 K=1,M
  CCC=BARR(J,K)
  DDD=XVEKT(K)
  QQQ=QQQ+CCC*DDD
66 CONTINUE
  BBB=ZVEKT(J)
  AAA=COVARR(I,J)
  EEE=AAA*(BBB-QQQ)
  AAASUM=AAASUM+EEE
65 CONTINUE
  IF(AAASUM.EQ.0.0) GOTO 64
  EENORM=EENORM+AAASUM*AAASUM
  XVEKT(I)=XVEKT(I)+AAASUM
64 CONTINUE
  IANT=IANT+1
  EENORM=DSQRT(EENORM)
  WRITE(3,67) IANT,EENORM
67 FORMAT(' -->working: iterativ refinement nr.',I2,': refinement
* norm: ',E8.2)
  IF(EENORM.GT.0.1) GOTO 68
  IF(IANT.EQ.20) GOTO 68
  IF(EENORM.LT.1E-7) GOTO 69
  GOTO 63
68 IF(EENORM.GT.1E-05) IERR=1
  IF(IERR.EQ.0) GOTO 69
  CALL LFEED(1)
  CALL TWRITE(' **>error: least square: iterative refinement
* failed',54)
  CALL LFEED(1)
  GOTO 70
69 CALL TWRITE(' -->working: least square: iterative refinement

```

```

* completed ',58)
C Q-----
70 AAASUM=0.0
  DO 71 K=1,M
  BBB=XVEKT(K)
  CCC=ZVEKT(K)
  AAASUM=AAASUM+BBB*CCC
71 CONTINUE
  QQQ=AAASUM
  QQQ=YHU-QQQ
  IF((N-M-1).LE.0) QQQ=0.0
  IF((N-M-1).GT.0) QQQ=QQQ/(N-M-1)
  IF(QQQ.GT.0.0) GOTO 72
  CALL LFEED(1)
  CALL TWRITE(' **>error : least square: negative variance ',46)
  CALL LFEED(1)
  QQQ=0.0
72 Q=QQQ
C COVARR-----
  DO 81 I=1,IB
  DO 81 J=1,IB
  IF((I.GT.M).OR.(J.GT.M)) COVARR(I,J)=0.0
  BBB=COVARR(I,J)
  BBB=QQQ*BBB
  COVARR(I,J)=BBB
81 CONTINUE
  CALL TWRITE(' -->working: least square: covarr. completed ',46)
  CALL TWRITE(' -->working: least square: completed',36)
  RETURN
END
C*****
C** MATRISEINVERTERING
C**
C*****
  SUBROUTINE INVERS(AARR,IB,N,AINV,DVEKT,IVEKT,IERR)
  REAL AARR(IB,IB),AINV(IB,IB),DVEKT(IB)
  INTEGER IVEKT(IB)
  DOUBLE PRECISION AAASUM,AAA,BBB,CCC,AAAMAX,AAALD
  CALL TWRITE(' -->working: matrix inversion: begin',36)
  C INITIALISERING-----
  M1=-1
  IANTIT=IERR
  DO 11 I=1,IB
  DVEKT(I)=0.0
  IVEKT(I)=0
  DO 11 J=1,IB
  AINV(I,J)=0.0
  11 CONTINUE
  ANORM=0.0
  DO 21 I=1,N
  DMAX=-1E8
  DO 22 J=1,N
  AHJELP=ABS(AARR(I,J))

```

```

      DMAX=AMAX1(DMAX,AHJELP)
      ANORM=ANORM+AAARR(I,J)*AAARR(I,J)
22  CONTINUE
      DVEKT(I)=DMAX
21  CONTINUE
      ANORM=SQRT(ANORM)/(N*N)
C K-LØKKE I LU-FAKTORISERING
DO 31 K=1,N
C BEREGN Mik-----
DO 32 I=K,N
      AAASUM=0.0
      KM1=K-1
      DO 33 J=1,KM1
        AAA=AAARR(I,J)
        BBB=AAARR(J,K)
        AAASUM=AAASUM+AAA*BBB
33  CONTINUE
      IF(K.EQ.1) AAASUM=0.0
      AAARR(I,K)=AAARR(I,K)-AAASUM
32  CONTINUE
C FINN PIVOTELEMENT-----
      AAMAX=-JF8
      DO 34 J=K,N
        ATEMP=ABS(AARR(J,K))/DVEKT(J)
        IF(AAMAX.GT.ATEMP) GOTO 34
        AAMAX=ATEMP
        IPK=J
34  CONTINUE
      IVEKT(K)=IPK
C SINGULÆRKONTROLL-----
      IF(AARR(IPK,K).NE.0.0) GOTO 38
      IERR=-1
      CALL LFEED(1)
      CALL TWRITE(' **>error: singular matrix',26)
      CALL LFEED(1)
      RETURN
C BYTTING AV KOLONNER-----
38 DO 35 J=1,N
      AHJELP=AAARR(K,J)
      DHJELP=DVEKT(K)
      AARR(K,J)=AARR(IPK,J)
      DVEKT(K)=DVEKT(IPK)
      AARR(IPK,J)=AHJELP
      DVEKT(IPK)=DHJELP
35  CONTINUE
C OPPDATERING AV Vk-----
      KPI=K+1
      KM1=K-1
      IF(K.EQ.N) GOTO 31
      DO 36 J=KPI,N
        AAASUM=0.0
        DO 37 I=1,KM1
          AAA=AAARR(K,I)
          BBB=AAARR(I,J)
          AAASUM=AAASUM+AAA*BBB
37  CONTINUE
      IF(K.EQ.1) AAASUM=0.0
      AARR(K,J)=(AARR(K,J)-AAASUM)/AARR(K,K)
      KPI=K+1
      DO 52 I=KPI,N
        AAASUM=0.0
        IM1=I-1
        DO 53 J=K,IM1
          AAA=AAARR(I,J)
          BBB=AAARR(J,K)
          AAASUM=AAASUM+AAA*BBB
53  CONTINUE
      AARR(I,K)=(AARR(I,K)-AAASUM)/AARR(I,I)
52  CONTINUE
51  CONTINUE
      AARR(N,N)=1/AARR(N,N)
C U-1*L-1-----
      DO 61 I=1,N
        DO 62 J=1,N
          AAASUM=0.0
          IF(I.EQ.1) AAASUM=0.0
          AARR(K,J)=(AARR(K,J)-AAASUM)/AARR(K,K)
36  CONTINUE
      C SLUTT MED K-LØKKE-----
31  CONTINUE
C OVERLESNING TIL U-----
      AINV(I,J)=0.0
      DO 45 I=1,N
        DO 45 J=1,N
          IF(I.EQ.J) AINV(I,J)=1.0
          IF(I.LT.J) AINV(I,J)=AARR(I,J)
45  CONTINUE
      CALL TWRITE(
        *' -->working: matrix inversion: LU-decomposition completed',57)
C U-1-----
      DO 41 K=2,N
        K=N-KK+2
        KM1=K-1
        DO 42 III=1,KM1
          I=KM1-III+1
          AAASUM=0.0
          IP1=I+1
          DO 43 J=IP1,K
            AAA=AINV(I,J)
            BBB=AINV(J,K)
            AAASUM=AAASUM+AAA*BBB
43  CONTINUE
          AINV(I,K)=-AAASUM
42  CONTINUE
41  CONTINUE
C OVERLESNING TIL L-----
      DO 55 I=1,N
        DO 55 J=1,N
          IF(I.LT.J) AARR(I,J)=0.0
55  CONTINUE
C L-1-----
      NM1=N-1
      DO 51 K=1,NM1
        AARR(K,K)=1/AARR(K,K)
        KPI=K+1
        DO 52 I=KPI,N
          AAASUM=0.0
          IM1=I-1
          DO 53 J=K,IM1
            AAA=AAARR(I,J)
            BBB=AAARR(J,K)
            AAASUM=AAASUM+AAA*BBB
53  CONTINUE
          AARR(I,K)=(AARR(I,K)-AAASUM)/AARR(I,I)
52  CONTINUE
51  CONTINUE
      AARR(N,N)=1/AARR(N,N)
C U-1*L-1-----
      DO 61 I=1,N
        DO 62 J=1,N
          AAASUM=0.0
```

```

DO 63 K=1,N
AAA=AINV(I,K)
BBB=AARR(K,J)
AAAASUM=AAAASUM+AAA*BBB
63 CONTINUE
DVEKT(J)=AAAASUM
62 CONTINUE
DO 64 J=1,N
AARR(I,J)=DVEKT(J)
64 CONTINUE
61 CONTINUE
C TILBAKEPIVOTERING-----
DO 71 KKK=1,N
K=N-KKK+1
IPK=IVEKT(K)
IF(IPK.EQ.K) GOTO 71
DO 72 I=1,N
AHJELP=AARR(I,K)
AARR(I,K)=AARR(I,IPK)
AARR(I,IPK)=AHJELP
72 CONTINUE
71 CONTINUE
C OVERLESNING I AINV-----
AINORM=0.0
DO 75 I=1,N
DO 75 J=1,N
AINV(I,J)=AARR(I,J)
AINORM=AINORM+AINV(I,J)*AINV(I,J)
75 CONTINUE
ANORM=ANORM*SQRT(AINORM)/(N*N)
WRITE(3,76) ANORM
76 FORMAT(' -->working: matrix inversion: condition number:',E7.1)
C SLUTT-----
99 CALL TWRITE(' -->working: matrix inversion: completed',40)
RETURN
END
C*****
C**
C** BEREGNING AV TIDSPUNKT I HAVMODELL
C**
C*****
FUNCTION TIDCAL(HOYD,TIDMIN,IORD,DVEKT)
REAL DVEKT(7)
IMIN=TIDMIN
DO 21 I=IMIN,15000,1000
TID1=I-1000
TID2=I
HOYD1=HOYCAL(TID1,IORD,DVEKT)
HOYD2=HOYCAL(TID2,IORD,DVEKT)
IF((HOYD.GE.HOYD2).AND.(HOYD.LE.HOYD1))
* .05.((HOYD.LE.HOYD2).AND.(HOYD.LE.HOYD1)) GOTO 21
DO 22 J=1,7
TIDM=(TID1+TID2)/2
HOYD1=HOYCAL(TID1,IORD,DVEKT)
HOYD2=HOYCAL(TID2,IORD,DVEKT)
HOYDM=HOYCAL(TIDM,IORD,DVEKT)
END
END
IF((HOYD.LE.HOYD2).AND.(HOYD.GT.HOYDM))
* .OR.((HOYD.GE.HOYD2).AND.(HOYD.LT.HOYDM))) GOTO 23
IF((HOYD.LE.HOYD1).AND.(HOYD.GT.HOYDM))
* .OR.((HOYD.GE.HOYD1).AND.(HOYD.LT.HOYDM))) GOTO 24
TID1=TIDM
GOTO 22
TID2=TIDM
GOTO 22
22 CONTINUE
TIDCAL=TIDM
RETURN
21 CONTINUE
TIDCAL=15000
RETURN
END
C*****
C**
C** BEREGNING AV SD PÅ HOYDE I HAVMODELL
C**
C*****
FUNCTION VARCAL(TID,IORD,COV)
REAL COV(7,7)
DOUBLE PRECISION AAADIF,BBB,CCC,DDD
TIDMIL=TID/1000.
AAADIF=0.0
DO 41 I=1,IORD
BBB=TIDMIL**I
DO 41 J=1,IORD
CCC=COV(I,J)
DDD=TIDMIL**J
AAADIF=AAADIF+BBB*CCC*DDD
41 CONTINUE
VARCAL=AAADIF
RETURN
END
C*****
C**
C** BEREGNING AV HØYDE I HAVMODELL
C**
C*****
FUNCTION HOYCAL(TID,IORD,DVEKT)
REAL DVEKT(7)
DOUBLE PRECISION AAAHOY,BBB,CCC
TIDMIL=TID/1000.
AAAHOY=0.0
DO 41 I=1,IORD
BBB=TIDMIL**I
CCC=DVEKT(I)
AAAHOY=AAAHOY+BBB*CCC
41 CONTINUE
HOYCAL=AAAHOY
RETURN
END

```



```
C*****
C**
C** BEREGNING AV FLATE-HØYDE
C**
C*****
FUNCTION ZHLCAL(X,Y,IHORD,H)
REAL ZHLCAL,XARR(5),YARR(5),H(5,5)
IH=IHORD+1
DO 20 K=1,IH
  XARR(K)=X**(K-1)
  YARR(K)=Y**(K-1)
20 CONTINUE
SUM=0.0
DO 21 I=1,IH
  DO 21 J=1,IH
    SUM=SUM+XARR(I)*H(I,J)*YARR(J)
21 CONTINUE
ZHLCAL=SUM
RETURN
END
C*****
C**
C** LINEÆRTRANSFORMASJON AV STRANDFORSKYVNINGSDATA
C**
C*****
SUBROUTINE LINTRA(HOYD,HOYD1,HOYD2,ZHL,ZHLVAR,ZNO,ZNOVAR)
HOYD=ZHL*ZNO
DIFF=SQRT(ZNO*ZNO*ZHLVAR+ZHL*ZHL*ZNOVAR+ZHLVAR*ZNOVAR)
HOYD1=HOYD-DIFF
HOYD2=HOYD+DIFF
RETURN
END
C*****
C**
C** LOGARITMETRANSFORMASJON AV STRANDFORSKYVNINGSDATA
C**
C*****
SUBROUTINE LOGTRA(HOYD,HOYD1,HOYD2,ZHL,ZHLVAR,ZNO,ZNOVAR)
ZHLOG=ALOG(ZHL+1)
HOYD=EXP(ZHLOG*ZNO)-1
DIFF=ZHLOG*ZHLOG*ZNOVAR+((ZNO*ZNO)/((ZHL+1)*(ZHL+1)))*ZHLVAR
DIFF=(HOYD+1)*SQRT(DIFF)
HOYD1=HOYD-DIFF
HOYD2=HOYD+DIFF
RETURN
END
```

```

*****
**
**      Subrutinebibliotek
**      for kartplotterutiner
**
*****
SUBROUTINE SDIINI foretar initialisering av
den grafiske hukommelsen.....s. A - 58

SUBROUTINE PARAME foretar utskrivning av
kartparametre.....s. A - 58

SUBROUTINE DATALES foretar innlesing av en datafil.....s. A - 59

SUBROUTINE EKVIDI foretar ekvidistanse-dialog.....s. A - 59

SUBROUTINE TOPGET foretar innlesing
fra topo-databasen.....s. A - 59

SUBROUTINE RUTGET foretar innlesing av 8-bits
tall fra to 10x10-ruter.....s. A - 59

SUBROUTINE RUTPUT foretar lagring av 8-bits
tall fra to 10x10-ruter.....s. A - 59

SUBROUTINE RUTMAL foretar lagring av 8-bits
10x10-rute-array.....s. A - 60

SUBROUTINE RUTLES foretar lesing av 8-bits
10x10-rute-array.....s. A - 60

SUBROUTINE ENKODE foretar lagring av
parametre på temporærfil.....s. A - 60

SUBROUTINE DEKODE foretar lesing av
parametre fra temporærfil.....s. A - 61

FUNCTION INTGET foretar innlesing av 8-bits
tall fra to 1x1-ruter.....s. A - 61

FUNCTION BYTGET foretar innlesing av 4-bits
tall fra 1x1-ruter.....s. A - 61

SUBROUTINE INTPUT foretar lagring av 8-bits
tall fra to 1x1-ruter.....s. A - 61

SUBROUTINE BYTPUT foretar lagring av 4-bits
tall fra 1x1-ruter.....s. A - 61

SUBROUTINE INTBYT foretar deling av 8-bits
tall i to 4-bits.....s. A - 61

```

```

*****
**
**      Subrutinebibliotek for kartplotterutiner
**
*****
**      Programert av Kjell Kjenstad
**      Desember 1982
**
*****
C
C
C*****
C** initialisering av SDI
C*****
SUBROUTINE SDIINI
COMMON IPAGE,IBASE,IOBS,IHMAX,IHMIN,IX(7),JY(7)
CALL WORKON(IPAGE)
CALL SCALE(1,378,1,241)
CALL UNSCAL
CALL RES(0)
CALL CLIP
CALL DISP(IPAGE)
CALL XAREA(7,130,18,241,0)
CALL XLINE(7,241,378,241,7)
CALL SCRON
CALL XAREA(7,1,16,129,0)
DO 21 J=1,15
CALL XAREA(10,8*J,16,8*J+6,J)
21 CONTINUE
RETURN
END
C*****
C** utskrift av parametre
C**
C*****
SUBROUTINE PARAME
COMMON IPAGE,IBASE,IOBS,IHMAX,IHMIN,IX(7),JY(7)
XMAAL=(1000*IOBS)/6.7
KPRUTE=2*JY(6)+IX(6)+1
KARUTE=2*JY(7)+IX(7)+1
WRITE(3,21) JY(5),XMAAL,IBASE,JY(4),IOBS,KPRUTE,JY(3)
* ,KARUTE,IHMIN,JY(2),IHMAX,JY(1),(IX(K),K=1,5)
21 FORMAT(I5,'-----~
* ' Kartmålestokken.....: 1:',F5.0,'000'//,
* ' ~ ~ ~ ~ ~ '
* ' Topobasennummer.....: ',I5//,
* I5,'-----~
* ' Antall 100 m mellom observasjoner: ',I5//,
* ' ~ ~ ~ 3 ~ 4 ~ ~ '
* ' Plotterute.....: ',I5//,
* I5,'-----~
* ' Arbeidsrute.....: ',I5//,
* ' ~ ~ 1 ~ 2 ~ ~ '
* ' Minste pakningsverdi.....: ',I5//,

```

```

* 15, '-----' ,
* ' Største pakningsverdi.....: ', I5/,
* ' ~ ~ ~ ~ ~ ' /,
* 15, '-----' /,
* ' 5I4, //)
WRITE(3,22)
22 FORMAT(' 0: Fortsett',/ ' 1: Tilbake til hovedmeny'
*,/ ' 2: Nye parametre')
NEXT=IREAD(' : ',2)
CALL TDV(25)
IF((NEXT.LE.0).OR.(NEXT.GT.2)) RETURN
GOTO (23,24),NEXT
23 CALL FCHAIN('MENYPRZCCOM',0)
24 CALL FCHAIN('INITIAZCCOM',0)
END
C*****
C**
C** lesing på datafil
C**
C*****
SUBROUTINE DATLES(ILISTE)
BYTE FILE(11)
COMMON IPAGE, IBASE, IOBS, IHMAX, IHMIN, IX(7), JY(7)
CALL FIREAD(FILE, '?XX? MAT')
CALL OPEN(8, FILE, 0)
IARUTE=IX(7)
JARUTE=JY(7)
IPRUTE=IX(6)
JPRUTE=JY(6)
IORIGO=IX(IARUTE+2)
JORIGO=JY(JARUTE+2)
IRANGE=((IX(IARUTE+3)-IX(IARUTE+2))*IOBS)/10
JRANGE=((JY(JARUTE+3)-JY(JARUTE+2))*IOBS)/10
I2=IORIGO+IRANGE
J2=JORIGO+JRANGE
I3=IORIGO
J3=JORIGO
DO 21 IA=1, ILOK
READ(8,22,END=24) X,Y
FORMAT(2F5.0)
X=X-IBASE*1000
IF((X.LT.I3).OR.(X.GT.I2).OR.(Y.LT.J3).OR.(Y.GT.J2))
* GOTO 21
I=X-IORIGO
J=Y-JORIGO
CALL BYTPUT(I,J,IPRUTE,JPRUTE,15)
21 CONTINUE
24 ENDFILE 8
RETURN
END
C*****
C**
C** lesing på topo-database
C**
C*****
SUBROUTINE TOPGET(IUNIT,I10,J10,JTOPO)
LOGICAL ITOPO(10,10),II(4)
INTEGER JTOPO(10,10)
COMMON IPAGE, IBASE, IOBS, IHMAX, IHMIN, IX(7), JY(7)
JREC=(IX(5)-IX(1))/10-1
IARUTE=IX(7)
JARUTE=JY(7)
IORIGO=IX(IARUTE+2)
JORIGO=JY(JARUTE+2)
IREC=(JREC+1)*((JORIGO-JY(1))/10+J10-1)+(IORIGO-IX(1))/10+I10+1
*,((ITOPO(I,J),I=1,10),J=1,10)
DO 21 I=1,10
DO 21 J=1,10
JTOPO(I,J)=ITOPO(I,J)+128
21 CONTINUE
RETURN
99 CALL TWRITE(' ***** READ-ERROR *****',31)
CALL LFEED(1)
RETURN
END
C*****
C**
C** hentning av 8 bits 10-rute på skjerm
C**
C*****
SUBROUTINE RUTGET(I10,J10,JJRUTE,ITOPO)
INTEGER I10,J10,JJRUTE,IOBS,ITOPO(10,10)
COMMON IPAGE, IBASE, IOBS, IHMAX, IHMIN, IX(7), JY(7)
DO 21 J=1,10
DO 21 I=1,10
ITOPO(I,J)=INTGET(10*(I10-1)+I,10*(J10-1)+J,JJRUTE)

```

```
21 CONTINUE
RETURN
END
C*****
C**
**
**
C** lagring av 8 bits 10-km rute på skjerm
**
C*****
SUBROUTINE RUTPUT(I10,J10,JJURUTE,I10PO)
INTEGER I10,J10,JJURUTE,IOBS,IKONST,I10PO(10,10)
COMMON IPAGE,IBASE,IOBS,IHMAX,IHMIN,IX(7),JY(7)
DO 21 J=1,10
DO 21 I=1,10
CALL INTPUT(10*(I10-1)+I,10*(J10-1)+J,JJURUTE,I10PO(I,J))
RETURN
END
C*****
C**
**
**
C** lagring av 8 bits 10-km ensfarget rute på skjerm
**
C*****
SUBROUTINE RUTMAL(IHAV)
INTEGER IHAV(19,13)
COMMON IPAGE,IBASE,IOBS,IHMAX,IHMIN,IX(7),JY(7)
IARUTE=IX(7)
JARUTE=JY(7)
IPRUTE=IX(6)
JPRUTE=JY(6)
IORIGO=IX(IARUTE+2)
JORIGO=JY(JARUTE+2)
IRANGE=(IX(IARUTE+3)-IX(IARUTE+2))*IOBS/10
J RANGE=((JY(JARUTE+3)-JY(JARUTE+2))*IOBS)/10
IMAX=(IRANGE*IOBS)/100+1
JMAX=(J RANGE*IOBS)/100+1
DO 22 I=1,IMAX
DO 22 J=1,JMAX
IXX=10*(I-1)+19
IYY=10*(J-1)+121
CALL XREAD(IXX,IYY,I8)
CALL XREAD(IXX+180,IYY,I4)
IHAV(I,J)=16*I8+I4
22 CONTINUE
DO 23 I=1,19
CALL XREAD(1,I,I8)
CALL XREAD(2,I,I4)
IHAV(I,13)=16*I8+I4
23 CONTINUE
DO 24 J=1,12
CALL XREAD(3,J,I8)
CALL XREAD(4,J,I4)
IHAV(19,J)=16*I8+I4
24 CONTINUE
RETURN
END
C*****
C**
**
**
C** LAGRING AV TEMPORÆR PARAMETERFIL
**
C*****
SUBROUTINE ENKODE
COMMON III(19)
CALL OPEN(6,'KARTPARATMP',0)
DO 21 I=2,19
WRITE(6,22) III(I)
22 FORMAT(IX,I10)
21 CONTINUE
ENDFILE 6
RETURN
END
C*****
```

```

C*****
C**
C** LESING AV TEMPORÆR PARAMETERFIL
C**
C*****
SUBROUTINE DEKODE
COMMON III(19)
CALL OPEN(6,'KARTPARATMP',0)
DO 21 I=2,19
  READ(6,22) III(I)
22 FORMAT(I10)
21 CONTINUE
ENDFILE 6
RETURN
END
C*****
C** funksjon for henting av 8 bits tall på skjerm
C**
C*****
FUNCTION INTGET(I,J,JRUTE)
INTEGER BYTGET,INTGET,JRUTE
INTGET=16*BYTGET(I,J,0,JRUTE)+BYTGET(I,J,1,JRUTE)
RETURN
END
C*****
C** funksjon for henting av 4 bits tall på skjerm
C**
C*****
FUNCTION BYTGET(I,J,IIRUTE,JRUTE)
INTEGER BYTGET,IIRUTE,JRUTE,IBYTE
CALL XREAD(IIRUTE*180+I+18,JRUTE*120+J,IBYTE)
BYTGET=IBYTE
RETURN
END
C*****
C** lagring av 8 bits tall på skjerm
C**
C*****
SUBROUTINE INPUT(I,J,JRUTE,IVERDI)
INTEGER JRUTE,IVERDI,IBYTE8,IBYTE4,IIRUTE8,IIRUTE4
CALL INTBYT(IVERDI,IBYTE8,IBYTE4)
CALL BYTPUT(I,J,0,JRUTE,IBYTE8)
CALL BYTPUT(I,J,1,JRUTE,IBYTE4)
RETURN
END
C*****
C** lagring av 4 bits tall på skjerm
C**
C*****
SUBROUTINE BYTPUT(I,J,IIRUTE,JRUTE,IBYTE)
INTEGER IIRUTE,JRUTE,IBYTE
CALL XDOT(IIRUTE*180+I+18,JRUTE*120+J,IBYTE)

```

```

RETURN
END
C*****
C** deling av 8 bits tall i to 4 bits tall
C**
C*****
SUBROUTINE INTBYT(IVERDI,IBYTE8,IBYTE4)
INTEGER IVERDI,IBYTE8,IBYTE4
HJELP=IVERDI
HJELP=HJELP/16
IBYTE8=HJELP
IBYTE4=(HJELP-IBYTE8)*16
RETURN
END

```

```

*****
**
**
**
**

```