

Litt ekstra hjelp for programmering av interpolasjons-øvingen

Denne oppgaven kan deles inn i flere deloppgaver:

1. Hovedprogrammet gjør innlesing og sjekking av en tabell av data som senere skal brukes for interpolasjonene
 - a. Deklarasjoner
 - b. Åpning av filer
 - c. Lesing av tabell (N linjer med XT og YT)
 - d. Sjekker at tabellen har minst 2 sett av data
 - e. Sjekker at tabellen er sortert i stigende orden, evt. kaller opp sorterings-subrutinen fra øving 5

2. Først leses antall tabellverdier, NN, og deretter NN verdier av X, for så å utføre interpolasjon for å finne motsvarende Y-verdier for 1.-, 2. og 3.-ordens interpolering, dvs. finn Y1, Y2, Y3 ved å kalle subrutinen LAGRANGE.

3. Til slutt skrives resultatene til utskriftsfilen. Ved å lagre resultatene i dimensjonerte variabler, Y1(I), Y2(I), Y3(I), kan utskriften gjøres utenfor løkken.

```
C----deklarasjoner
REAL XT(100),YT(20),X(20),Y1(20),Y2(20),Y3(20)
INTEGER N,I,M,NN
CHARACTER SORTERT*1
C-----åpning av filer
OPEN(UNIT=11,FILE="INTERP.INN",STATUS="OLD")
OPEN(UNIT=12,FILE="INTERP.UT",STATUS="UNKNOWN")
C-----koden nedenfor leser tabellen og sjekker
READ(11,*)N
IF(N.LT.2)THEN
PRINT * "N MAA VAERE STOERRE ENN 1"
STOP
ENDIF
READ(11,*)(XT(I),YT(I),I=1,N)
SORTERT="J"
DO 10 I=1,N-1
IF(X(I).GT.X(I+1))THEN
SORTERT="N"
ENDIF
GO TO 11
10 CONTINUE
11 IF(SORTERT="N")THEN
CALL SORTERING(N,XT,YT)
ENDIF
```

```
READ(11,*)NN
DO 100 I=1,NN
READ(11,*)X(I)
M=1
CALL LAGRANGE(X(I),Y1(I),M,N,XT,YT)
M=2
CALL LAGRANGE(X(I),Y2(I),M,N,XT,YT)
M=3
CALL LAGRANGE(X(I),Y3(I),M,N,XT,YT)
```

```
WRITE(12,200)(X(I),Y1(I),Y2(I),Y3(I),I=1,NN)
200 FORMAT (4F10.4)
```

Interpoleringsrutinen LAGRANGE

Denne rutinen kalles fra hovedprogrammet, med tabellen (N,XT, YT), X-argumentet (X), interpoleringsgrad (M), og resultatparameteren (Y) i argumentlisten. Oppgavene som skal gjøres i subrutinen er:

1. Først sjekkes at interpoleringsgraden ikke er større en antall tabellpunkter tilsier.

```
IF(M.GE.N)THEN
WRITE(12,210)
210 FORMAT(" FEIL I INNDATA: M=",I2," N=",I2)
STOP
ENDIF
```

2. Deretter må intervallet for interpoleringen fastlegges, dvs. vi må finne ut hvor i tabellen $XT(I) < X < XT(I+1)$, og dermed bestemme hvilke tabellpunkter som skal benyttes i interpoleringen.

3. Så kan selve interpoleringen gjøres ved å benytte Lagrange-formelen oppgitt i øvingsteksten:

$$y = \frac{(x - x_2)(x - x_3) \dots (x - x_N)}{(x_1 - x_2)(x_1 - x_3) \dots (x_1 - x_N)} y_1 + \frac{(x - x_1)(x - x_3) \dots (x - x_N)}{(x_2 - x_1)(x_{21} - x_3) \dots (x_2 - x_N)} y_2 + \dots + \frac{(x - x_1)(x - x_2) \dots (x - x_{N-1})}{(x_N - x_1)(x_N - x_2) \dots (x_N - x_{N-1})} y_N$$

Inn- og ut-filer

Innfilen, INTERP.INN, inneholder først selve tabellen (NN verdier av XT og YT), og deretter antall (N) og alle X-verdiene som skal interpoleres for. Innfilen for datasett 1 vises til høyre

```
C----sett maksimumsverdier i tilfelle X > XT(N)
  IB=N
  IA=IB-M
C--finn laveste (IA) og høyeste tabellindeks (IB)
  DO 10 I=1,N-1
  IF(X.LT.XT(I))THEN
    IA=I-1
C----juster punktet hvis M>1
  IA=IA-M/2
  IB=IA+M
C----sjekk at IA ikke er lavere enn 1 eller IB høyere enn N
  IF(IA.LT.1)THEN
    IA=1
  IB=IA+M
  ELSEIF(IB.GT.N)THEN
    IB=N
    IA=IB-M
  ENDIF
  GO TO 11
  ENDIF
10 CONTINUE
11 CONTINUE
```

```
DO 100 J=IA,IB
  P(J)=1.0
  DO 100 I=IA,IB
  IF(I.NE.J)THEN
    P(J)=P(J)*(X-XT(I))/(XT(J)-XT(I))
  ENDIF
100 CONTINUE
  Y=0.
  DO 200 I=IA,IB
  Y=Y+P(I)*YT(I)
```

```
6
0.2      0.008
1.0      1.0
0.4      0.064
0.6      0.216
0.        0.
0.8      0.512
11
0.
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1.0
```

Utfilen, INTERP.UT, inneholder inndata samt interpolerte verdier, og utfilen for datasett 1 vises til høyre

INNLEST TABELL, N= 6

X	Y
.2000	.0080
1.0000	1.0000
.4000	.0640
.6000	.2160
.0000	.0000
.8000	.5120

SORTERT TABELL, N= 6

X	Y
.0000	.0000
.2000	.0080
.4000	.0640
.6000	.2160
.8000	.5120
1.0000	1.0000

INTERPOLERTE VERDIER

X	Y1	Y2	Y3
.0000	.0000	.0000	.0000
.1000	.0040	-.0020	.0010
.2000	.0080	.0080	.0080
.3000	.0360	.0300	.0270
.4000	.0640	.0640	.0640
.5000	.1400	.1280	.1250
.6000	.2160	.2160	.2160
.7000	.3640	.3460	.3430
.8000	.5120	.5120	.5120
.9000	.7560	.7320	.7290
1.0000	1.0000	1.0000	1.0000