REGNECENTRALEN

DANSK INSTITUT FOR MATEMATIKMASKINER

UOFFICIEL SEKVENS

SEKVENS-BETEGNELSE
Peter Naur nr. 4. 2. december 1959

2. udgave 11. februar 1960

INTERPOLATION, DASK-TAL,
I EN TABEL MED ÆKVIDISTANT ARGUMENT

Indhop	Indgang	Udgang
0A8 16 n A 00	C(AR) = $p \times 2^{-11} + \frac{x-x_o}{d} \times 2^{-10}$ Tabel: $C(p-2a) = f(x_o-ad)$ .... $C(p-2) = f(x_o-d)$ $C(p) = f(x_o)$ $C(p+2) = f(x_o+d)$ .... $C(p+2b) = f(x_o+bd)$ DASK-ta .	f(x) til AR og MR idet der ved interpolationen er benyttet n punkter, som er centreret så godt som muligt omkring x.

Kodelængde 0 - (49 + 2n)	Undersekvenser ingen
Begyndelsesadresse lige	Arbejdsceller I sekvensen: 45A8 - (18+2n A8
Grundparametre ingen	Permanente konstanter ingen
Programparametre n A 00	Stop i 31A8: 31A851 ved internt spild.

Sekvensen kan også benyttes til interpolation nær begyndelsen eller slutningen af en tabel på følgende måde:

1) Adressen for den første tabelværdi, som ønskes benyttet, lagres i 11A8adr
2) Denne adresse (i ARvadr) - C(AR) som defineret ovenfor til AR
3) Indekshop til 7A8, programparameter nA00:

7 A8 16
n A 00

1. Metode. Der benyttes en recursiv metode, der er kendt som Neville's metode (se W.E. Milne: Numerical Calculus, side 72).

2. Den uafhængige variable specificeres ved indhoppet gennem C(AR). Det tal som skal anbringes her fås ved følgende transformation:

$$C(AR) = p \times 2^{-11} + \frac{x - x_0}{d} \times 2^{-10}$$

hvor $f(x_0)$ er lagret i hec p, og f(x) er tabelleret med interval d i x. Betydningen af denne transformation ses hvis man betragter det specielle tilfælde at $2^{-11}p = x_0$ og $d = 2^{-10}$ (dvs. at der i hec h netop er lagret funktionsværdien til adressen h opfattet som DASK-tal). Da haves C(AR) = x.

3. Tabellens længde tages ikke i betragtning ved interpolationen. Det påhviler brugeren at sørge for at de værdier, som specificeres for x og n, ikke løber ud over tabellens grænser. Sekvensen bruger altid de n på hinanden følgende tabelværdier, som så godt som muligt centreres af x.

4. Nøjagtighed. Interpolationsbrøkdelen kan kun opgives med 29 binaler og man vil derfor aldrig opnå bedre nøjagtighed end hvad der svarer hertil i funktionen f(x). Fejlen vil imidlertid normalt være større end dette idet processen naturligvis negligerer led som afhænger af differenser af tabellen som er af ordenen n og større.

5. Stop. Sekvensen stopper på ordre 31A8: 31A851 hvis en bestemt indre operation fører til en division af et større tal med et mindre. Det er temmelig indviklet at give en fuldstændig regel for hvornår dette vil ske, da det både afhænger af tabelværdierne og n, men som hovedregel kan anføres at faren opstår når n multipliceret med differensen mellem på hinanden følgende tabelværdier bliver større end ca. 1. Brug derfor ikke et unødig stort n - n=4 - 6 vil i de allerfleste tilfælde være passende.

6. ALGOL program: (NB: Ikke korrekt ALGOL 60).

```
procedure Neville (x , f[ ], n) =: (fx); integer (n); array(f[1+x-n/2:x+n/2]);
begin array (C[0:n-1]); integer (xo, q, v, i);
Neville:  xo := 1+x-n/2 ;
          A  := xo - x ;
          for q := 0(1)n-1 ;
          begin  C[q]  := f[xo + q]  ;
                 if (q = 0) ; go to P ;
                 A := A + 1 ;
                 B := A ;
                 v := 0 ;
                 for i := q-1(-1)0 ;
                 begin  v := v + 1 ;  B := B - 1 ;
                        C[i]  := (AxC[i] - BxC[i+1])/v end for i ;
          P: end for q ;
          fx := C[0]  ;
          return end Neville;
```

```
Indhop ->           0     46 A8 08   Gem argument
                    1      1 D  21   - n
                    2     44 A8 20   + 2
                    3     29 A  0F   11A8adr := 2 x entier (p/2 + 1 - n)
                    4     29 A  0C
                    5     11 A8 29
                    6     46 A8 01
Specielt indhop ->  7     46 A8 08   A := x0 - x
                    8     42 A8 34   Gem IRB
                    9     10 A8 14   10A8adr := 0
40 ->              10    (0)A   35   IRB := q
                   11    (0)B   40   C[q] := f [q + x0]
                   12     50 B8 08
                   13     15 A8 33   Hop for IRB = q ≠ 0
                   14     37 A8 10   Hop for q = 0
13 ->              15     44 A8 60   A := A + 1
                   16     46 A8 06
                   17     48 A8 08   B := A
                   18     45 A8 68   v := 0
36 ->              19   2046 B  35   for IRB := q-1 step -1 until 0 do
                   20     45 A8 66   v := v + 1
                   21     44 A8 61   B := B - 1
                   22     48 A8 06
                   23     46 A8 44
                   24     50 B8 0A
                   25     50 B8 08
                   26     48 A8 45                     C IRB  :=
                   27     52 B8 0A
                   28     50 B8 06                     (A x C IRB
                   29     45 A8 62     Spild
                   30     50 B8 03     test             - B x C IRB + 2 )/v
                   31     31 A8 51
                   32     50 B8 40
                   33     45 A8 2B
                   34      0 A  07
                   35     50 B8 08
                   36     19 A8 33   end for IRB
14 ->              37     10 A8 66   q := q + 2
                   38      1 A  0D
                   39      1 D  21   - n
                   40     10 A8 51   end for q
                   41     50 A8 44   AR := MR := C 0
                   42    (0)A   35   Retabler IRB
                   43      2 D  10   Udhop
                   44      2 A  00   2, 15, 21,
                   45                v, 18, 20, 29, 33
                46 47                argument, A, 0, 6, 7, 16, 23
                48 49                B, 17, 22, 26
                50 51                C 0
                52 53                C 1
```