Literatur-Nachtrag

Bezugsform-Nr.: SB12-0032-0

Sachcode: 22

Nachtrag-Nr.: SN12-0482

Datum: August 1977

Vorhergehende Nachträge: keine

IBM 5100 MATHEMATIK/APL Benutzerhandbuch

Dieser Nachtrag bezieht sich auf die oben aufgeführte Broschüre.
Die Änderungen sind am Rande durch einen vertikalen Strich gekennzeichnet.

Bitte folgende Seiten

entfernen:	einordnen:
Titelblatt	Titelblatt
1 und 2	1 und 2
47 und 48	47 und 48
91 bis 96	91 bis 96
101 und 102	101 und 102
129 und 130	129 und 130
133 und 134	133 und 134
145 bis 148	145 bis 148
151 und 152	151 und 152
157 bis 160	157 bis 160
169 und 170	169 und 170
179 und 180	179 bis 182

IBM Deutschland GmbH

Inhalt

(8) Die Eigenvektoren werden ermittelt und ausgegeben:

```
MATRIX DER EIGENVEKTOREN
  7.071068E−01  −6.324555E−01  −3.162278E−01  −1.574395E−16
 −7.071068E−01  −6.324555E−01  −3.162278E−01  −1.487116E−16
 −2.266233E−17  −3.162278E−01   6.324555E−01   7.071068E−01
  0.000000E−01  −3.162278E−01   6.324555E−01   7.071068E−01
```

(9) Man kann die Resultate auf Band speichern.

```
RESULTATAUSGABE AUF DATEI? (0=N, 1=J)
□:
   [0]
```

(10) Nachdem die Aufgabe gelöst ist kann man wählen, ob eine neue Aufgabe gelöst werden soll, oder ob zur MENU-Routine zurückgekehrt werden soll.

```
SIND WEITERE AUFGABEN ZU LOESEN? (1=J, 0=N)
□:
   [0]
```

Bemerkungen

(1) Aufbau der Eingabedatei (bei Bandeingabe)

Überschrift

Matrix A: Ordnung der Matrix

Matrix B

(2) Eingabe über Tastatur (bei Bandeingabe)
Art des zu lösenden Problemes: AX = BXL oder ABX = XL.

(3) Aufbau der Ausgabedatei (bei Bandausgabe)

Überschrift

Ordnung der Matrix

Eigenwerte

Matrix X der Eigenvektoren

Fehlernachrichten

EINGB AUSSERH. ZUL BEREICH.———— : ————. WIEDERHOLEN.
NICHT ZUL. EINGABE GEBR. WERTE. WIEDERHOLEN. Die Dimensionen für die Matrizen sind fehlerhaft (Tastatureingabe). Neuer Versuch möglich.

FALSCHE DIMENSION (Banddatei-Eingabe). Die Matrix hat falsche Dimensionen. Abbruch der Arbeit.

MATRIX B NICHT POSITIV DEFINIT. Die Matrix B ist (möglicherweise auf Grund von Abrundungsfehlern) nicht positiv definit. Abbruch der Arbeit.

EIGENWERTE NACH 20 ITERATIONEN NICHT GEFUNDEN. Bestimmung der Eigenwerte mißlungen, die größtmögliche Zahl der Iterationen wurde erreicht. Die bei der 20. Iteration gefundenen Werte werden als Eigenwerte betrachtet.

Literaturnachweis

Martin R. S. und Wilkinson J. H. "Reduction of the Symmetric Eigenproblem Ax = 1Bx and Related Problems to Standard Form" Numerische Mathematik, Band 11, 1968, Seite 99–110.

Wilkinson J. H., Reinsch C. und Martin R. S.: "Householder's Tridiagonalization of a Symmetric Matrix". Numerische Mathematik, Band 11, 1968, Seite 181–195.

Wilkinson J. H. und Martin R. S. "The Implicit QL Algorithm", Numerische Mathematik, Band 12, 1968, Seite 377–383.

Dubrulle A. A. "A Short Note on the Implicit QL Algorithm for Symmetric Tridiagonal Matrices", Numerische Mathematik, Band 15, 1970, Seite 450.

(3) Die Anzahl der Funktionswerte wird eingegeben. Die maximale Anzahl ist abhängig von der Maschinengröße.

```
ANZAHL DER PUNKTE IN TABELLE EINGEBEN (2≤M≤60)

□:

    6
```

(4) Es muß zwischen equidistanten und nicht equidistanten Abszissenwerten gewählt werden. Wenn nicht equidistante Abszissenwerte gewählt wurden, müssen die Abszissenwerte in monoton aufsteigender Reihenfolge mit einem Wert für jeden Funktionswert eingegeben werden. Für equidistante Abszissenwerte wird Anfangswert und Schrittweite verlangt.

```
'E' FUER EQUIDISTANTE ABSZISSE EINGEBEN

E

ABSZISSEN-ANFANGSWERT EINGEBEN

□:

    0

ABSZISSEN-SCHRITTWEITE EINGEBEN

□:

    .1
```

(5) Die angegebene Zahl von Funktionswerten muß eingegeben werden. Ebenso muß die Anzahl der Basis-Funktionen eingegeben werden.

```
6 FUNKTIONSWERTE EINGB.

□:

    0 .021 .081 .179 .321 .499

EXAKTE ZAHL VON BASIS-FUNKTIONEN EINGB. (MAX = 6)

□:

    3
```

(6) Die relative Toleranz und die maximale Anzahl von Iterationsschritten wird eingegeben.

```
RELATIVE TOLERANZ BEI MAX ABWEICHG EINGB. (1E−14≤TOL<1)

□:

    1E−6
```

```
MAXIMALZAHL VON ITERATIONEN EINGEBEN (MAX = 20)

□:

     20
```

(7) Das Programm zeigt das Kennwort der Aufgabe, Anzahl der Tabellenpunkte und Anzahl der Basis-Funktionen an.

```
6 TABELLENPKTE: 3 BASIS-FUNKTIONEN

AUFG. KENNW: AMM EX
```

(8) Das Programm bestimmt die Koeffizienten und gibt die Ergebnisse aus.

```
MAXIMALE ABWEICHUNG IST 0.001

ANZAHL VON ITERATIONEN IST 9

FUNKTION     KOEF DER APPROX FUNKT.
1            0.000000E-01
2            0.000000E-01
3            2.000000E00
```

(9) Wenn es gewünscht wird, können die Daten mit Toleranz und maximalen Iterationen verarbeitet werden.

```
WOLLEN SIE WEITERE AUFGABEN LOESEN (1=J, 0=N)

□:

     0
```

Bemerkungen

(1) Eingabe von Datei (wenn anwendbar)

Überschrift
Anzahl der Tabellenpunkte
Zeichen, das die zu verarbeitende Tabellenart anzeigt.
Ein Wert 'E' zeigt an, daß die Tabelle equidistante Abszissenwerte hat.
Andernfalls irgendein Zeichen, das nicht ein 'E' enthält.
Im Falle von equidistanten Abszissen:
Anfangswert und Schrittweite der Abszisse
Werte der Funktion an den Tabellenpunkten
Im Falle von nicht equidistanten Abszissenwerten:
Die Werte der Abszisse
Die Werte der Funktion an den Tabellenpunkten.

Fehlermeldungen

FALSCHE ZAHL. . WIEDERHLN. Die Anzahl von einzugebenden Tabellenpunkten war nicht in dem Bereich von $2 \leqslant M \leqslant 60$(32 K) oder $2 \leqslant M \leqslant 90$ (48K, 64K)

FALSCHE SCHRITTW. . WIEDERHLN. Eine Null-Schrittweite wurde eingegeben.

ITERATIONSZAEHLER UEBER 20. Die Maximalzahl von Iterationen wurde erreicht, bevor eine optimale Lösung des linearen Programms gefunden wurde. Der Benutzer kann zusätzliche Iterationen anfordern.

KEINE ENDLICHE LOESUNG. VERARBEITUNG BEENDET.

Literaturhinweise

Barrodale, I., and Young, A., "Algorithms for Best L, and L Linear Approximation on a Discrete Set", Numerische Mathematik, Vol. 8, 1966, 295–306.

SLS: Glättung einer Tabellenfunktion

Aufgabe

Lokale Glättung einer Tabellenfunktion durch die Methode des kleinsten Quadrates (Polynom vom Grade 1 relativ zu drei benachbarten Punkten).

Methode

Wenn x_i ein Abszissenwert in der Tabelle und y_i ein dazugehöriger Funktionswert ist, ist der entsprechende Wert z_i der geglätteten Funktion gegeben durch

$$z_i = \overline{y}_i + m_i (x_i - \overline{x}_i),$$

wobei

$$\overline{x}_i = \frac{1}{3} \sum_{j=i-1}^{i+1} x_j, \quad \overline{y}_i = \frac{1}{3} \sum_{j=i-1}^{i+1} y_j$$

$$m_i = \frac{\sum_{j=i-1}^{i+1} (x_j - \overline{x}_i)(y_j - \overline{y}_i)}{\sum_{j=i-1}^{i+1} (x_j - \overline{x}_i)^2}$$

An den Endpunkten der Untertabelle, auf denen die Operation durchgeführt wird, werden die Werte x_i, y_i, und m_i für die nächsten Tabellenpunkte benutzt. Wenn die Abszissenwerte equidistant sind, werden vereinfachte Formeln benutzt.

Verfahren

Dieses Beispiel glättet die Funktion $y=x^2$, tabelliert für sechs Werte an equidistanten Abszissenwerten mit Anfangswert 0 und Schrittweite 0.05.

(1) Im ersten Schritt wählen Sie die Eingabeart aus. Bei manueller Eingabe wird als erstes ein Kennwort der Aufgabe verlangt.

```
SLS . . .  LOKALE GLAETTUNG EINER TABFKT DURCH METH.
           DER KLEINSTEN QUADRATE (POLYNOM VOM
           GRAD 1 RELATIV ZU DREI PKTE)

WOLLEN SIE DATEN ÜBER TASTATUR EINGEBEN (0=N, 1=J)

□:

    [1]

KENNWORT D. AUFGABE?

[SLS EX]
```

(2) Anzahl der Funktionswerte wird eingegeben. Die Maximalzahl ist abhängig von der Maschinengröße.

```
ZAHL VON PKTE IN EINGABE TAB EINGB. BEREICH: 3–200
□:
   [6]
```

(3) Es muß zwischen equidistanten und nicht equidistanten Abszissenwerten gewählt werden. Wenn nicht equidistante Abszissenwerte gewählt werden, müssen die Abszissenwerte in monoton aufsteigender Reihenfolge mit einem Wert für jeden Funktionswert eingegeben werden. Für equidistante Abszissenwerte wird Anfangswert und Schrittweite verlangt.

```
ABSZISSENWERTE EQUIDISTANT? (0=NEIN, 1=JA)
□:
   [1]
ABSZISSEN-ANFANGSWERT EINGEBEN
□:
   [0]
ABSZISSEN-SCHRITTWEITE EINGEBEN
□:
   [.05]
```

(4) Die angegebene Zahl von Funktionswerten muß eingegeben werden.

```
6 FUNKTIONSWERTE EINGEBEN
□:
   [0 .0025 .01 .0225 .04 .0625]
```

(5) Das Programm glättet die Funktion und gibt die Ergebnisse aus.

AUFG. KENNW.: SLS EX

UNTER TAB EINTRAG	ABSZISSE X [I]	WERT DER FUNKTION [I]	GEGLAETT. WERT [I]	DIFFERENZ Y [I] – S [I]
1	0.000E–00	0.000E–00	–8.333E–04	8.333E–04
2	5.000E–02	2.500E–03	4.167E–03	– 1.667E–03
3	1.000E–01	1.000E–02	1.167E–02	– 1.667E–03
4	1.500E–01	2.250E–02	2.417E–02	– 1.667E–03
5	2.000E–01	4.000E–02	4.167E–02	– 1.667E–03
6	2.500E–01	6.250E–02	6.167E–02	8.333E–04

(6) Sie haben die Möglichkeit,die Ergebnisse auf Datei zu speichern.

```
WOLLEN SIE ERGEB. IN EINE DATEI SPEICHERN (0=N, 1=J)

□:

  [0]
```

(7) Nachdem Ihre Aufgabe gelöst ist, können Sie weitere Aufgaben lösen oder zu dem Programm MENU zurückkehren.

```
WOLLEN SIE WEITERE AUFGABEN LOESEN? (0=NEIN, 1=JA)

□:

  [0]
```

Bemerkungen

(1) Eingabe von Datei (wenn anwendbar)

Für equidistante Abszissenwerte:

Überschrift
Anzahl der Funktionswerte
Buchstabe 'E', der equidistante Punkte anzeigt
Anfangswert und Schrittwerte der Abszisse
Werte der Funktion

Für nicht equidistante Abszissenwerte:

Überschrift
Anzahl der Funktionswerte
Irgendein Buchstabe wie 'N', der nicht 'E' enthält
Werte der Abszisse
Werte der Funktion

(2) Ausgabe auf Datei (wahlweise)

Für equidistante Abszissenwerte:

Überschrift
Zahl der geglätteten Werte
Buchstabe 'E'
Anfangswert und Schrittweite der Abszisse
Werte der geglätteten Funktion

Für nicht equidistante Abszissenwerte:

Überschrift
Anzahl von geglätteten Werten
Buchstabe 'N'
Werte der Abszisse
Werte der geglätteten Funktion

FMN: Lokales Minimum einer Funktion mehrerer Variabler

Aufgabe

Die Bestimmung eines lokalen Minimus einer Funktion von mehreren Variablen.

Methode

n sei die Anzahl der Variablen $\{d_i;\ i=1,...,n\}$ eine Menge von linear unabhängigen Einheitsvektoren und $y=f(x_i,...,x_n) = F(X)$ die Funktion für die ein lokales Minimum gesucht wird.

Der Algorithmus basiert auf einem iterativen Prozeß. Eine Iteration besteht aus den folgenden Schritten:

Startend von einem Punkt X_0 wird die Funktion sukzessiv entlang allen Richtungen d_i minimiert:

bestimme t, den Wert von s, der

$F(X_{i-1}+sd_i)$ minimiert und bilde $i=1,...,n$

$$X_i=X_{i-1}+td_i$$

Startend von X_n wird die Funktion entlang der Richtung (X_n-X_0) minimiert. Dies führt zu X_{n+1}, einer neuen Annahme von Koordinaten des Minimums.

$$\text{Ersetze } d_n \text{ durch } \frac{X_n - X_o}{\|X_n - X_o\|_2}$$

Verfahren

Dieses Beispiel minimiert die Funktion:

$$f=x_1+X_2+\frac{1}{X_1}+\frac{1}{x_2}$$

mit dem Anfangswert $x_1=.9$ und $x_2=1.1$

Bevor Sie FMN ausführen, müssen Sie eine Benutzerroutine in dem Arbeitsspeicher definieren. MENU suspendiert sich selber, so daß der Benutzer eine Funktion definieren kann. Der Benutzer gibt dann MENU ein, um FMN zu beginnen. In diesem Beispiel ist die Benutzerroutine SFMN wie folgt definiert:

```
    ∇F ← SFMN X
[1]  F ← (x/X)++/÷X
    ∇
```

(1) Zuerst wird der Name der Unterroutine, die die Funktion berechnet, eingegeben.

```
FMN ..  LOKALES MINIMUM EINER FUNKTION MEHRER VARIABLER

NAME DER ROUTINE EINGEBEN, DIE FUNKTION BERECHNET

SFMN
```

(2) Die Anzahl der Variablen in der Funktion und ihre Anfangswerte werden angegeben.

```
ANZAHL DER VARIABLEN IHRER FUNKTION EINGEBEN
(MAXIMUM 15)

□:

  2

2 ANFANGSWERTE FUER DIE VARIABLEN EINGEBEN

□:

  .9 1.1
```

(3) Die Maximalzahl der durchzuführenden Iterationen wird eingegeben.

```
MAXIMALZAHL DURCHZUFUEHRENDER ITERATIONEN EINGEBEN
(MAX 20)

□:

  20
```

(4) Das Programm löst die Aufgabe und gibt die Ergebnisse aus.

```
ANZAHL V. ITERATIONEN:       5
WERT DER FUNKTION AN DEM MINIMUM: 3
VARIABLE          ANF. KOORD.      KOORD FUER MINIMUM
1                 9.000000E-01     9.999998E-01
2                 1.100000E00      1.000000E00

EXECUTE-TASTE DRUECKEN
```

(5) Sie können weitere Aufgaben lösen oder zu dem MENU-Programm zurückkehren.

```
WOLLEN SIE EINE ANDERE AUFG. LOESEN (0=N, 1=J)

□:

  0
```

Bemerkungen

(1) Der Benutzer wird daran erinnert, daß FMN das lokale Minimum der angegebenen Funktion(en) findet. Die Lösung hängt in höchstem Maße von den Anfangswerten des Versuchs ab.

(2) Wenn kein Minimum gefunden wird, kann es drei Gründe geben. 1. Die Funktion hat vielleicht kein Minimum. 2. Die angenommenen Anfangswerte x_1, x_2 usw. sind vielleicht so schlecht, daß der Code das lokale Minimum nicht findet, obwohl es existiert. 3. Das Minimum ist numerisch vielleicht schlecht definiert.

(2) Die Argumente für die Funktionen müssen eingegeben werden.

```
WERT(E) DES ARGUMENTS EINGEBEN
1 BIS 200 WERTE ERL.

□:

    1E6 100 25 1 –1
```

(3) Die Funktionen werden berechnet und die Ergebnisse ausgegeben.

```
FRESNEL INTEGRALE

     ARGUMENT       INTEGRAL C(X)      INTEGRAL S(X)
     1.000000E06    5.000000E–01       5.000000E–01
X≥   1000000       INTEGRAL (E) BERECHNET FUER UNENDL. X
     1.000000E02    4.796285E–01       4.657020E–01
     2.500000E01    4.878799E–01       4.212170E–01
     1.000000E00    7.217059E–01       2.475583E–01
     –1.000000E00   7.217059E–01       2.475583E–01
X<   0. ABSOL. WERT BENUTZT
```

(4) Nachdem Ihre Aufgabe gelöst ist, können Sie weitere Aufgaben lösen oder zu dem Programm MENU zurückkehren.

```
WOLLEN SIE WEITERE AUFGABEN LOESEN (0=N, 1=J)

□:

    0
```

Bemerkungen

Dateieingabe/-ausgabemöglichkeiten bestehen nicht.
Je nach Wahl berechnet FIS die Werte von S (x), C (x) oder von beiden.
Bis zu 200 Werte des Arguments können für eine Aufgabe eingegeben werden.
Die Argumente sollten in dem Bereich $0 \leqslant x < 10^6$ für S (x) und C (x) liegen.

Fehlermeldungen

FALSCHER CODE. BITTE WIEDERHLN. Falscher Code für die Operationswahl. Dem Benutzer werden weitere Versuche ermöglicht.

INTEGRAL (E) FUER UNENDL. X BERECHNET. $|x| \geqslant 10^6$.

X < 0 ABSOL. WERT BENUTZT. $x < 0$. $|x|$ wird für die Berechnung der Integrale benutzt.

Literaturhinweise

IBM Procedure Library-Mathematics (PL-MATH) Program Reference Manual (SH20-0985) 1971, Procedure FICS/FIC/ FIS.

SCI: Sinus- und Cosinusintegrale

Aufgabe

Berechnung von Sinus- und Cosinusintegralen Si (x) und Ci (x) definiert durch:

$$Si(x) = \int_0^x \frac{\sin t}{t} dt,$$

$$Ci(x) = \gamma + \ln x + \int_0^x \frac{\cos t - 1}{t} dt$$

wobei $\gamma = 0.577\ldots$ die Euler'sche Konstante ist.

Methode

Die Werte der Integrale erhält man durch die folgenden rationalen Approximationen:

1. Berechnung von Ci (x) und Si (x) für $x > 13$

- Bereich $x > 13$
  Die Sinus- und Cosinusintegrale erlauben die Darstellung:

$$Si(x) = \frac{\pi}{2} - \frac{13}{x} \cdot (P(x) \cos x + Q(x) \sin x)$$

$$Ci(x) = \frac{13}{x} \cdot (P(x) \sin x - Q(x) \cos x)$$

Mit $z = \frac{169}{x^2}$ werden die folgenden rationalen Approximationen für P und Q benutzt:

$$P(x) = \frac{\sum_{v=0}^{6} a_v z^v}{\sum_{v=0}^{6} b_v z^v} + \epsilon(x) P(x)$$

mit $\max|\epsilon(x)| \leqslant 4 \cdot 10^{-18}$.

$$Q(x) = \sqrt{z}\,[\,R(x) + \epsilon(x) R(x)\,]$$

wobei

$$R(x) = \frac{\sum_{v=0}^{7} a_v z^v}{\sum_{v=0}^{6} b_v z^v}$$

und $\max|\epsilon(x)| \leqslant 5.6 \cdot 10^{-18}$.

- Bereich $3 < x \leqslant 6$ (enthält Null von Ci(x))

$$Ci(x) = \ln\left(\frac{x}{x_0}\right) + (x - x_0)(x + x_0)\,[\,R(x) + \epsilon(x)\,R(x)]$$

wobei $x_0 = 3.384180422551186$,

$$R(x) = \frac{\sum_{v=0}^{6} a_v (1-z)^v}{\sum_{v=0}^{5} b_v z^v}, \quad z = \frac{x^2}{36}$$

und $\max|\epsilon(x)| \leqslant 3.2 \cdot 10^{-18}$.

- Bereich $0.5 < x \leqslant 3$ (enthält Null von Ci(x))

$$Ci(x) = \ln\left(\frac{x}{x_0}\right) + (x - x_0)(x + x_0)\,[R(x) + \epsilon(x)\,R(x)]$$

wobei $x_0 = 0.6165054856207162$,

$$R(x) = \frac{\sum_{v=0}^{4} a_v (1-z)^v}{\sum_{v=0}^{5} b_v z^v}, \quad z = \frac{x^2}{9}$$

und $\max|\epsilon(x)| \leqslant 4.3 \cdot 10^{-19}$.

- Bereich $0 < x \leqslant 0.5$

$$Ci(x) = \gamma + \ln x + \frac{x^2}{4}\,[R(x) + \epsilon(x)\,R(x)]$$

wobei $\gamma = 0.5772156\ldots$ (Euler'sche Konstante)

$$R(x) = \frac{\sum_{v=0}^{3} a_v z^v}{\sum_{v=0}^{2} b_v z^v}, \quad z = x^2$$

und $\max|\epsilon(x)| \leqslant 1.3 \cdot 10^{-19}$.

Negative Werte von x werden durch Symmetrieüberlegungen abgedeckt.

Die relative Genauigkeit der Ergebnisse liegt in der Größenordnung der Maschinengenauigkeit, außer in der Umgebung der Nullstellen der Funktion.

Verfahren

Dieses Beispiel berechnet die Sinus- und Cosinusintegrale für sechs Argumente: 10^6, 100, 25, 1, −1, −100.

(1) Als erstes wird die gewünschte Funktion angegeben.

```
SCI . . .   BERECHG. VON SINUS- UND COSINUSINTEGRALEN

CODE FUER DIE GEWUENSCHTEN INTEGRALE EINGEBEN
'S' FUER SINUS 'C' FUER COSINUS, 'B' FUER BEIDE

B
```

(2) Die Argumente werden eingegeben.

```
ANZAHL DER ARGUMENTWERTE EINGEBEN (MAX 200)

□:

    6

6 ARGUMENTWERTE EINGEBEN

□:

    1E6 100 25 1 -1 -100
```

(3) Die Funktionen werden berechnet und die Ergebnisse ausgegeben.

```
SINUS- UND COSINUSINTEGRALE

      ARGUMENT        SINUSINTEGRAL     COSINUSINTEGRAL
      1.000000E06     1.570796E00       0.000000E-01
|X|>  1000000 INTEGRAL (E) BERECHNET FUER UNENDL. ARGUMENT
      1.000000E02     1.562225E00       -5.148825E-03
      2.5000000E01    1.531483E00       -6.848597E-03
      1.000000E00     9.460831E-01       3.374039E-01
     -1.000000E00    -9.460831E-01       3.374039E-01
X<0.INTEGRAL BERECHNET FUER |X|
     -1.000000E02     1.562225E00           -5.148825E-03
X<0. INTEGRAL BERECHNET FUER |X|
```

(4) Nachdem Ihre Aufgabe gelöst ist, können Sie weitere Aufgaben lösen oder zu dem Programm MENU zurückkehren.

```
WOLLEN SIE WEITERE AUFGABEN LOESEN (0=NEIN, 1=JA)

□:

    0
```

Bemerkungen

Dateieingabe/-ausgabemöglichkeiten sind nicht verfügbar.
Je nach Wahl berechnet SCI Werte des Integrals Si(x), Integral Ci(x) oder von beiden.
Bis zu 200 Werte des Arguments können für eine Aufgabe eingegeben werden.
Die Argumente sollten in dem Bereich $0 \leqslant x < 10^6$ für Si(x) und in dem Bereich $0 < x < 10^6$ für Ci(x) liegen.

3. Berechnung von $Y_1(x)$ für $x \leqslant 8$

In den folgenden Bereichen wird eine spezielle Approximation für die Berechnung von $\ln(x/x_0)$ in der Umgebung von x_0 benutzt, um eine relative Genauigkeit zu erhalten.

- Bereich $4 < x \leqslant 8$ (enthält Nullstelle von $Y_1(x)$ bei $x = x_0$ siehe weiter unten) 2x1

$$Y_1(x) = \frac{2}{x} \ln\left(\frac{x}{x_0}\right) \cdot J_1(x) + \frac{1}{x}(x - x_0)(x + x_0)\,[R(x) + \epsilon(x) \cdot R(x)]$$

wobei $x_0 = 5.429681040794135$

$$R(x) = \frac{\sum_{v=0}^{7} a_v (1-z)^v}{\sum_{v=0}^{7} b_v z^v}, \quad z = \frac{x^2}{64}$$

und $\max |\epsilon(x)| \leqslant 1.8 \cdot 10^{-18}$.

- Bereich $1 < x \leqslant 4$ (enthält Nullstelle von $Y_1(x)$ bei $x = x_0$ siehe weiter unten)

$$Y_1(x) = \frac{2}{\pi} \ln\left(\frac{x}{x_0}\right) \cdot J_1(x) + \frac{1}{x}(x - x_0)(x + x_0)\,[R(x) + \epsilon(x)\, R(x)]$$

wobei $x_0 = 2.197141326031017$

$$R(x) = \frac{\sum_{v=0}^{6} a_v z^v}{\sum_{v=0}^{5} b_v z^v}, \quad z = \frac{x^2}{16}$$

und $\max |\epsilon(x)| \leqslant 3.1 \cdot 10^{-19}$.

- Bereich $0 < x \leqslant 1$

$$Y_1(x) = \frac{2}{\pi}\left[\ln x \cdot J_1(x) - \frac{12}{x}\right] + x \cdot [R(x) + \epsilon(x)\, R(x)]$$

wobei

$$R(x) = \frac{\sum_{v=0}^{4} a_v z^v}{\sum_{v=0}^{3} b_v z^v}, \quad z = 4x^2$$

und $\max |\epsilon(x)| \leqslant 6.9 \cdot 10^{-20}$.

Negative Werte des Arguments werden durch Symmetriebeziehungen gehandhabt.

Verfahren

In diesem Beispiel werden Bessel-Funktionen der ersten und zweiten Art für die Ordnung 0 bis 4 mit Argument 9 berechnet.

(1) Zuerst wird die Funktionsart, die berechnet werden soll, ausgewählt.

```
BFS . . .  BESSEL-FUNKTIONEN ERSTER UND ZWEITER ART VON
           GANZZAHLIGER ORDNUNG

WOLLEN SIE FN. ERSTER ART BERECHNEN?  (0=NEIN, 1=JA)

□:

    [1]
```

(2) Die höchste Ordnung und das Argument werden angegeben.

```
MAXIMALE ORDNUNG EINGEBEN. GANZE ZAHL: 0≤N<200

□:

    [4]

ARGUMENT X EINGEBEN. 0≤1X<1000000

□:

    [9]
```

(3) Das Programm berechnet die Funktionen und gibt die Ergebnisse aus.

```
BESSEL-FUNKTION ERSTER ART

ARGUMENT = 9

ORDNUNG     FUNKTIONSWERT
0           −9.033361E−02
1             2.453118E−01
2             1.448473E−01
3           −1.809352E−01
4           −2.654708E−01
```

(4) Wenn eine andere Aufgabe gelöst werden soll, wird der Funktionstyp eingegeben.

```
WOLLEN SIE WEITERE AUFGABEN LOESEN?  (0=NEIN, 1=JA)

□:

    [1]

BFS. . BESSEL-FUNKTIONEN ERSTER UND ZWEITER ART VON
       GANZZAHLIGER ORDNUNG

WOLLEN SIE FN. ERSTER ART BERECHNEN?  (0=NEIN, 1=JA)

□:

    [0]
```

(5) Die maximale Ordnung und das Argument werden angegeben.

```
MAXIMALE ORDNUNG EINGEBEN. GANZE ZAHL: 0≤N<200

□:

   [4]

ARGUMENT X EINGEBEN 1E-6≤1X<1000000

□:

   [9]
```

(6) Die Funktionen werden errechnet und die Resultate ausgegeben.

```
BESSEL-FUNKTION ZWEITER ART

ARGUMENT = 9

ORDNUNG        FUNKTIONSWERT
0                2.499367E-01
1                1.043146E-01
2               -2.267557E-01
3               -2.050949E-01
4                9.002576E-02
```

(7) Wenn alle Aufgaben gelöst worden sind, geht man zu dem Programm MENU zurück.

```
WOLLEN SIE WEITERE AUFGABEN LOESEN (0=NEIN, 1=JA)

□:

   [0]
```

Bemerkungen

Dateieingabe/-ausgabemöglichkeiten gibt es nicht.
Die Ordnung der Funktion muß eine ganze Zahl aus dem Bereich $0 \leqslant N \leqslant 199$ sein. BFS gibt n+1 Werte entsprechend den Ordnungen 0, 1, . . . , N für jede Aufgabe zurück.
Für Bessel-Funktionen der ersten Art sollte x in dem Bereich $10^{-6} < x < 10^6$ liegen. Für Bessel-Funktionen der zweiten Art sollte das Argument x in dem Bereich $x < 10^6$ liegen. In dem Fall eines negativen Arguments x, wird die Funktion für 1 x 1 berechnet.
Für einige Argumente können Funktionswerte zu groß oder zu klein werden, um in der Maschine dargestellt zu werden. Wenn dies auftritt, werden Sie einen Bereichsfehler bekommen und die Verarbeitung wird beendet.

Fehlermeldungen

IHRE EINGABE LIEGT AUSSERHALB 0 BIS 199. WIEDERHLN. Falsche Ordnung der Funktion: die gegebene Ordnung muß eine ganze Zahl aus dem Bereich $0 \leqslant N \leqslant 199$ sein. Dem Benutzer werden weitere Versuche ermöglicht.

AUSSERHALB.WIEDERHLN. Ungültiges Argument: gültige Argumente liegen in dem Bereich $10^{-6} < x < 10^{6}$ für Funktionen erster Art und $x < 10^{6}$ für Funktionen zweiter Art. Dem Benutzer werden weitere Versuche ermöglicht.

FUNKTION ZWEITER ART WIRD FUER |X| BERECHNET. $x < 0$ für Funktion zweiter Art.

Literaturhinweise

Blanch, G., "Numerical Evaluation of Continued Fractions," SIAM Review, Vol. 6, No. 4, 1964, 383–421.

(5) Die Funktionen werden berechnet und die Resultate ausgegeben.

```
ARGUMENT = 1.1

ORDNUNG        FUNKTIONSWERT
-.50000         3.450746E-01
-1.50000       -9.916930E-01
-2.50000        2.359543E-00
-3.50000       -9.733500E-00
```

(6) Wenn alle Aufgaben gelöst wurden, werden Sie zu dem Programm MENU zurückkehren.

```
WOLLEN SIE WEITERE AUFGABEN LOESEN? (0=NEIN, 1=JA)

□:
   [0]
```

Bemerkungen

Dateieingabe/-ausgabemöglichkeiten bestehen für BFF nicht.
Der absolute Wert der Ordnung der Funktion muß kleiner als 200 sein.
BFF gibt n+1 Funktionswerte der Ordnung i+a oder –(i+a), i=0, . . . ,n für eine Aufgabe zurück.
Das Argument x muß in dem Bereich $10^{-6} < x < 57$ liegen.
Für einige Argumente können die Funktionswerte für hohe Ordnungen zu groß oder zu klein für die Darstellung in der Maschine werden. Wenn dies auftritt, werden Sie einen Bereichsfehler bekommen.

Fehlermeldungen

ORDNUNG AUSSERHALB. WIEDERHLN. Die maximale Ordnung liegt außerhalb des gültigen Bereichs $i+a < 200$. Dem Benutzer werden weitere Versuche ermöglicht.

X AUSSERHALB. WIEDERHLN. Das Argument liegt außerhalb des gültigen Bereichs $10^{-6} < x < 57$. Dem Benutzer werden weitere Versuche ermöglicht.

Literaturhinweise

Cautschi, W., "Computational Aspects of Three-Term Recurrence Relations," SIAM Review, Vol. 9, No. 1, 1967, 24–82.

BMS: Modifizierte Bessel-Funktionen ganzzahliger Ordnung

Aufgabe

Berechnung der modifizierten Bessel-Funktionen positiver ganzzahliger Ordnung der ersten oder zweiten Art.

- Modifizierte Bessel-Funktionen erster Art:

  $I_j(x)$, J=0,1, ...,n

- Modifizierte Bessel-Funktionen zweiter Art:

  $K_j(x)$, j=0,1,...,n.

Methode

Die Werte der modifizierten Bessel-Funktion erster und zweiter Art für die Ordnungen 0 und 1 werden durch rationale Approximationen berechnet. Die Werte für andere Ordnungen erhält man mit Hilfe der Rückwärts-Rekursionsrelation.

Die relative Genauigkeit der Ergebnisse liegt in der Ordnung der Maschinengenauigkeit, außer in der Umgebung der Nullstellen der Funktion. Informationen über die verschiedenen benutzten Algorithmen werden weiter unten gegeben.

- Erste Art und Ordnung größer als 1.

  Die Werte der modifizierten Bessel-Funktionen $I_N(x)$ erhält man mit Hilfe der Rückwärts-Rekursionsrelationen.

  Die Gleichungen $G_n = I_n/I_{n-1}$ genügen:

  $$G_n = \frac{1}{\frac{2n}{x} + G_{n+1}} \qquad (1)$$

  G_{N+1} erhält man aus dem Kettenbruch:

  $$G_{N+1} = \frac{1}{\frac{2(N+1)}{x}+} \frac{1}{\frac{2(N+2)}{x}+} \cdots + \frac{1}{\frac{2(N+k)}{x}} + \cdots \qquad (2)$$

  Aus G_{N+1} erhält man die Werte $G_N, G_{N-1}, \ldots, G_1$ durch sukzessive Anwendung der Formel (1).

  Schließlich werden die Werte $I_1, I_2, \ldots, I_N$ berechnet mit:

  $$I_k = G_k \cdot I_{k-1} \text{ für } k = 1, 2, \ldots, N,$$

  mit dem Anfangswert $I_0(x)$ (siehe rationale Approximation weiter unten)

- Zweite Art und Ordnung größer als 1.

  Die Rekursionsrelation:

  $$K_{n+1}(x) = \frac{2n}{x} K_n(x) - K_{n-1}(x), \text{ für } n = 1, 2, \ldots, N-1$$

  wird benutzt, beginnend mit den Werten $K_0(x)$ und $K_1(x)$.

2. Berechnungen von $K_1(x)$

Bereich $0 < x \leqslant 2$

$$K_1(x) = I_1(x) \cdot \ln\left(\frac{x}{2}\right) + \frac{1}{x} - \frac{x}{4} + \frac{x}{2} \cdot [R(x) + \epsilon(x) R(x)]$$

wobei

$$R(x) = \frac{\sum\limits_{v=0}^{4} a_v(1-z)^v}{\sum\limits_{v=0}^{4} b_v z^v}, \quad z = \frac{x^2}{x}$$

und $\max|\epsilon(x)| \leqslant 2.7 \cdot 10^{-17}$.

Bereich $2 < x \leqslant 8$

$$K_1(x) = \frac{e^{-x}}{\sqrt{x/2}} [R(x) + \epsilon(x) R(x)]$$

wobei

$$R(x) = \frac{\sum\limits_{v=0}^{6} a_v z^v}{\sum\limits_{v=0}^{6} b_v z^v}, \quad z = \frac{2}{x}$$

und $\max|\epsilon(x)| \leqslant 1.2 \cdot 10^{-18}$.

Bereich $x < 8$

$$K_1(x) = \frac{e^{-x}}{\sqrt{x/8}} [R(x) + \epsilon(x) R(x)]$$

wobei

$$R(x) = \frac{\sum\limits_{v=0}^{5} a_v z^v}{\sum\limits_{v=0}^{5} b_v z^v}, \quad z = \frac{8}{x}$$

und $\max|\epsilon(x)| = 3.8 \cdot 10^{-18}$.

Negative Werte von x werden durch Symmetrierelationen abgedeckt.

Verfahren

Dieses Beispiel berechnet modifizierte Bessel-Funktionen erster und zweiter Art für das Argument 4.4 für die Ordnung 0 bis 5.

(1) Die Art der Funktion wird ausgewählt.

```
BMS . .   MODIFIZIERTE BESSEL-FUNKTIONEN ERSTER UND
          ZWEITER ART GANZZAHLIGER ORDNUNG

WOLLEN SIE FN. ERSTER ART BERECHNEN? (0=NEIN, 1=JA)

□:

    [1]
```

(2) Die maximale Ordnung und das Argument werden angegeben.

```
MAX. ORDG. EINGB. GANZE ZAHL: 0 ≤ N < 200

□:

    [5]

ARGUMENT X EINGEBEN 0 ≤ 1X < 170

□:

    [4.4]
```

(3) Die Funktionen werden berechnet und ausgegeben.

```
MODIFIZIERTE BESSEL-FUNKTION ERSTER ART

ARGUMENT = 4.4

ORDNUNG         FUNKTIONSWERT
0               1.601044E01
1               1.404622E01
2               9.625789E00
3               5.295504E00
4               2.404648E00
5               9.234161E–01
```

(4) Wenn eine andere Aufgabe gelöst werden soll, wird die Funktionsart eingegeben.

```
WOLLEN SIE WEITERE AUFGABEN LOESEN? (0=NEIN, 1=JA)

□:

    [1]

WOLLEN SIE FN. ERSTER ART BERECHNEN? (0=NEIN, 1=JA)

□:

    [0]
```

(5) Die maximale Ordnung und das Argument werden angegeben.

```
MAXIMALE ORDNUNG EINGEBEN. GANZE ZAHL: 0≤N<200

□:

    [5]

ARGUMENT X EINGEBEN. 1E−6≤1X<170

□:

    [4.4]
```

(6) Die Funktionen werden berechnet und die Ergebnisse ausgegeben.

```
MODIFIZIERTE BESSEL-FUNKTION ZWEITER ART

ARGUMENT = 4.4

ORDNUNG        FUNKTIONSWERT
0              7.149111E−03
1              7.923253E−03
2              1.075059E−02
3              1.769652E−02
4              3.488220E−02
5              8.111870E−02
```

(7) Wenn alle Aufgaben gelöst wurden, werden Sie zu dem Programm MENU zurückkehren.

```
WOLLEN SIE WEITERE AUFGABEN LOESEN? (0=NEIN, 1=JA)

□:

    [0]
```

Bemerkungen

Dateieingabe/-ausgabemöglichkeiten bestehen nicht.

Die Ordnung der Funktion muß eine ganze Zahl in dem Bereich von $0 \leq N < 200$ sein. BMS wird N + 1 Werte für jede Aufgabe zurückgeben, entsprechend den Ordnungen 0,1,...,N.

Für modifizierte Bessel-Funktionen erster Art sollte das Argument X in dem Bereich $X < 170$ liegen. Für modifizierte Bessel-Funktionen zweiter Art sollte das Argument X in dem Bereich $10^{-6} < X < 170$ liegen. Im Falle $X < 0$ wird die Funktion für $|X|$ berechnet.

Für einige Argumente können die Funktionswerte für hohe Ordnungen zu groß oder zu klein für die Darstellung in der Maschine werden. Wenn dieses auftritt, werden Sie einen Bereichsfehler erhalten. Die Verarbeitung wird beendet.

Fehlermeldungen

IHRE EINGABE LIEGT AUSSERHALB 0 BIS 199. WIEDERHLN. Ungültige Ordnung der Funktion: die gegebene Ordnung muß eine ganze Zahl in dem Bereich von $0 \leq N < 200$ sein. Dem Benutzer werden weitere Versuche ermöglicht.

X AUSSERHALB. WIEDERHLN. Ungültiges Argument: gültige Argumente liegen in dem Bereich $X < 170$ für Funktionen erster Art und in dem Bereich $10^{-6} < X < 170$ für Funktionen zweiter Art. Dem Benutzer werden weitere Versuche ermöglicht.

FUNKTION ZWEITER ART WIRD FUER |X| BERECHNET. $X < 0$, Funktionen zweiter Art werden für $|X|$ berechnet.

Literaturhinweis

Blanch, G., "Numerical Evaluation of Continued Fractions," SIAM Review, Vol. 6, No. 4, 1964, 383–421.

3. Eingabe von Datei (wenn anwendbar)
Überschrift
Anzahl der Nebenbedingungen, m
Anzahl der Variablen, n
Matrix A von Koeffizienten der Nebenbedingungen, mxn Matrix.

Matrix B der Schranken, (m+n) x2 Matrix. Die ersten m Reihen dieser Matrix enthalten die unteren und oberen Schranken, die zu den Nebenbedingungen gehören. Die letzten n Reihen von B sind die unteren und oberen Schranken der Variablen. Das erste Element jeder Reihe ist die untere Schranke und das zweite die obere Schranke.

Die Koeffizienten der Zielfunktion (n Elemente).

4. Ausgabe auf Datei (wahlweise)
Überschrift
Anzahl der Nebenbedingungen m
Anzahl der Variablen n
Der Wert der Zielfunktion
eine (m+n) x2 Matrix und (nx2) Paare von Werten x_j, d_j und
mx2 Paare von Werten y_i, p_i bei der optimalen Lösung.

Fehlermeldungen

UNMOEGLICHE SCHRANKE F. (VARIABLE/NEBENBED) NR. N. Ungültige Schranken in den Nebenbedingungen oder Beschränkungen der Variablen ($b_{i,1} < 10^{70}$ S/B $> 10^{70}$). Die Verarbeitung wird beendet.

LOESUNG UNBESCHRAENKT, VIELLEICHT WEGEN RUNDUNGSFEHLER. EINSCHR. NR. NICHT ERFUELLT. Der Ursprung des Problems liegt an einer Nebenbedingung, deren Index von der Routine ausgedruckt wird.

UNBESCHR. LOESUNG VARIABLE NR. N. Eine Variable hat keine definierte untere oder obere Grenze. Der Index der Variablen wird gedruckt und die Verarbeitung beendet.

UNMOEGLICHE SCHRANKE F. (VARIABLE/NEBENBED) NR. N. Eine Schranke einer Nebenbedingung oder einer Variablen, die nicht möglich ist, wurde entdeckt. Der Index der Nebenbedingung oder Variablen wird ausgedruckt und die Verarbeitung beendet.

DERZEITIGER STAND DER RECHNUNG. Beim Auftreten einer der obigen Bedingungen wird der derzeitige Status der Lösung angezeigt.

Literaturhinweise

IBM-Veröffentlichungen:
A Preface to Linear Programming and Its Applications (GE20-0350)

Introduction to Linear Programming (GE20-8171).

Anhang A APL-Fehlernachrichten

Elementarfunktionen, definierte Funktionen, Systemanweisungen, Systemvariable und Ein-/Ausgabe-Operationen können fehlerhaft verlaufen und somit Fehlernachrichten geben. Die folgende Liste enthält die Fehlernachrichten zusammen mit möglichen Ursachen und den empfohlenen Reaktionen des Benutzers.

Fehlernachricht	Ursache	Reaktion des Benutzers
ALREADY MARKED	Die angegebene Datei wurde früher schon formatiert.	Soll die Datei reformatiert werden, tippen Sie GO ein. *Bemerkung:* Alle Daten in den Dateien, die auf die letzte reformatierte Datei folgen, sind nicht mehr verfügbar.
CHARACTER ERROR	Ein ungültiges Zeichen ist eingetippt worden.	Tippen Sie die korrigierte Anweisung ein.
DEFN ERROR	Das Umschalten in die Arbeitsweise "Funktionsdefinition" war ungültig:	
	• Das Zeichen ∇ wurde fälschlicherweise in einer Anweisung benutzt. • Es wurde versucht, für eine verdeckte Funktion die Definition wieder zu öffnen.	Sollte die Funktion mit dieser Anweisung beginnen oder enden, so darf ∇ nur in der Anfangs- bzw. Endposition stehen.
	• Es wurde versucht, für eine Funktion die Definition wieder zu öffnen, dabei wurde mehr als nur der Funktionsname eingegeben.	Tippen Sie die korrigierte Anweisung ein.
	• Es wurde versucht, die Definition für eine neue Funktion zu öffnen, dabei wurde der Name einer früher definierten globalen Variablen benutzt.	Verwenden Sie einen anderen Funktionsnamen, oder löschen Sie die globale Variable.
	• Eine ungültige Änderung im Modus "Funktionsdefinition" wurde versucht.	Tippen Sie eine gültige Anweisung zum Ändern ein.
	• Es wurde versucht, eine hängende Funktion zu ändern.	Kann die Ausführung der unterbrochenen Funktion beendet werden, so löschen Sie den Statusanzeiger (siehe Kapitel 7) und ändern dann die Funktion.
DEVICE NOT OPEN	Eine nicht geöffnete Datei sollte gelesen werden.	Weisen Sie die Information, die zum Öffnen der Datei benötigt wird, der gemeinsamen Variablen zu.

Anhang B

DATEI-NR.	PROGRAMM-NAME	DATEI-TYP	DATEI-GRÖSSE	
01	MENU	7	006	
02	SPRM	8	003	
03	DISP	8	005	
04	SOL	8	008	
05	MCE	8	017	
06	MRE	8	013	
07	MRV	8	012	
08	MBE	8	015	
09	MSE	8	010	
10	MSG	8	015	
11	QTP	8	007	
12	QSP	8	006	INHALT PROGRAMM-BAND 1
13	QTI	8	005	
14	QSI	8	004	
15	DTQ	8	008	
16	QAR	8	004	
17	QGS	8	004	
18	QGSDATA	8	003	
19	ALI	8	008	
20	ACS	8	007	
21	APC	8	011	
22	POS	8	003	
23	APF	8	012	
24	AMM	8	011	
25	SLS	8	008	
26	FRZ	8	003	
27	FMN	8	008	
28	PCZ	8	011	
29	DER	8	009	
30	FFT	8	011	
31	POV	8	004	
32	ELS	8	012	INHALT PROGRAMM-BAND 2
33	GAM	8	003	
34	FIS	8	013	
35	SCI	8	013	
36	EXI	8	011	
37	BFS	8	016	
38	BFF	8	005	
39	BMS	8	015	
40	BMF	8	006	
41	MLP	8	019	
*42	EDITAPL	7	006	

*Siehe Anhang C.

Anhang C APL-Bibliothekswartung

Da die Bibliotheksprogramme auf der Bandkassette in Datenform stehen, müssen sie, bevor Änderungen vorgenommen werden können, in den Hauptspeicher der IBM 5100 gelesen und in eine APL-Funktion umgewandelt werden. Eine Ausnahme bildet die Datei, die das Programm MENU enthält. Dies liegt in ausführbarer Form vor.
Ein Dienstprogramm, das die Programme in den Hauptspeicher liest, in APL-Funktionen umwandelt und sie wieder im richtigen Format auf die Bandkassette schreibt, liegt vor. Es ist in der Datei EDITAPL gespeichert, die die letzte Datei der zweiten Bandkassette der Bibliotheksprogramme ist.
Soll die Datei benutzt werden, muß sie geladen werden:
)LOAD Dateinummer EDITAPL
Dateinr. ist für MATH/APL 19.
Der Arbeitsbereich enthält zwei Funktionen: EDITIN und EDITOUT.
Beide werden im folgenden besprochen.

EDITIN

Mit EDITIN werden APL-Bibliotheksprogramme von der Bandkassette in den Arbeitsbereich übertragen. Ist EDITIN erfolgreiche abgeschlossen, können die APL-Bibliotheksprogramme wie andere APL-Funktionen verändert und verbessert werden.
Als Programmeingabe wird die Dateinr. des gewünschten Bibliotheksprogramms gefordert. EDITIN fordert auch noch, daß vor der Ausführung die richtige Bandkassette eingelegt ist.

Sobald eine Routine eingelesen ist, wird ihr Name angezeigt.

Mit EDITIN kann das MENU-Programm nicht eingelesen werden. Diese Datei kann direkt mit dem APL-Ladebefehl eingelesen werden.

```
      EDITIN
EDITIN..INPUT ROUTINE FOR EDITING APL
        (Eingabefunktion für APL-Programmwartung)
ENSURE INPUT FILE ON DRIVE 1, HIT EXECUTE KEY
(Eingabedatei auf Einheit 1?  EXEC-Taste drücken)
ENTER INPUT FILE NUMBER
(Nr. der Eingabedatei eingeben)
 ⎕:

                    14
SUBR
BFS1
BFSF
BFSO
EVAL
SLOG
BFSS

IF YOU WISH TO CHANGE THE FILE NAME ENTER )DROP 14
BEFORE EXECUTING EDITOUT
```

EDITOUT

Die Funktion EDITOUT schreibt Bibliotheksfunktionen in dem Format auf die Bandkassette, das für die Funktion MENU erforderlich ist.
Die Funktionen werden in die *gleiche* Datei geschrieben, von der sie mit EDITIN gelesen wurden. Soll der Dateiname geändert werden, muß folgendes vor Ausführung von EDITOUT eingegeben werden:
)DROP Dateinr. Dateiname

EDITOUT fordert, daß das richtige Ausgabeband vor dem Beschreiben eingelegt ist. Als einzige Eingabe wird der Name der Ausgabedatei verlangt.

EDITOUT
EDITOUT..OUTPUT ROUTINE FOR EDITING APL
(Ausgabefunktion für APL-Programmwartung)
ENSURE OUTPUT FILE ON DRIVE 1, HIT EXECUTE KEY
(Ausgabedatei auf Einheit 1? EXEC-Taste drücken)
ENTER OUTPUT FILE NAME
(Name der Ausgabedatei eingeben)

BFS

SUBR
BFS1
BFS0
BFSF
SLOG
EVAL
BFSS

EDITOUT COMPLETED

Bemerkungen zur Benutzung der EDITAPL-Routinen

1. Wenn Bibliotheksfunktion geändert wird, sollte vorher eine Kopie der unveränderten Bänder gemacht und solange aufgehoben werden, bis man sicher ist, daß die Veränderungen laufen.

2. Die in EDITOUT benutzte Ausgabedateinr. ist die gleiche, die als Eingabedateinr. in EDITIN eingegeben wird.

3. $\underline{\Delta}$ und $\underline{\Delta}$IN sind globale Variable im EDITAPL Arbeitsbereich und dürfen *nicht* geändert werden.

4. *Alle* Funktionen des Arbeitsbereichs (außer EDITIN und EDITOUT) werden bei der Ausführung von EDITOUT auf die Bandkassette geschrieben. Zur Einsparung von Platz auf der Datei sollten alle unnötigen Funktionen im Arbeitsbereich gelöscht werden, bevor EDITOUT ausgeführt wird und zwar folgendermaßen:

   ⎕EX 'Funktionsname'

5. Ist eine Funktion auf das Band geschrieben, wird sie im Arbeitsbereich gelöscht. Deshalb können EDITIN und EDITOUT erneut ausgeführt werden.

6. Wenn EDITIN oder EDITOUT laufen, sollten keine anderen Funktionen ausgeführt werden. Bei der Ausführung von Funktionen werden die APL Symboltabelle und der Arbeitsbereich gefüllt. Dadurch kann für EDITIN und EDITOUT unmöglich werden, Funktionen von oder auf das Band zu bringen.

7. Nach mehreren Ausführungen der Funktionen EDITIN und EDITOUT kann die Nachricht 'SYMBOL TABLE FULL' (volle Symboltabelle) oder 'WS FULL' (voller Arbeitsbereich) erscheinen. Zum Weitermachen muß jetzt der EDITAPL-Arbeitsbereich neu geladen werden.

8. Wird die Funktion MLP mit EDITIN und EDITOUT geändert, muß eine IBM 5100 mit mindestens 48 K zur Verfügung stehen.

9. Da QGSDATA eine Datei ist, die konstante Daten und nicht Programme enthält, kann sie nicht mit EDITIN und EDITOUT gelesen und geschrieben werden.