

[04_04]九州大学大型計算機センター広報表紙奥付等

<https://hdl.handle.net/2324/1467977>

出版情報：九州大学大型計算機センター広報. 4 (4), 1971-08-02. 九州大学大型計算機センター
バージョン：
権利関係：

ライブラリプログラムの紹介

No. 250 C3/QU/F/DRPGMA

登録年月日 昭和46年6月28日

Reciprocal of Gamma Function

実変数ガンマ関数の逆数

| | | |
|----------------|---|---------------------------------------|
| 作 成 | 作 成 者 塩 川 浩 三 | 作成年月日 昭和46年5月1日 |
| 形 式 | a. コンプリートプログラム d. 手続き | b. サブルーチン c. 関数 e. 関数手続き |
| 使 用 言 語 | a. FORTRAN d. PL/I | b. ALGOL c. FASP e. その他 () |
| 使 用 機 種 | FACOM 230-60 | |
| 使用メモリ数 | a. コア(0.2)K語 | b. ディスクパック()K語 c. その他() |
| 使 用 機 器 構 成 | a. カードリーダー d. 紙テープリーダー f. 磁気テープ()ユニット g. ディスクパック h. その他() | b. ラインプリンタ e. 紙テープパンチ c. カードパンチ |
| 利用者の義務 | a. プログラム名と作成者名を明記する | b. 明記する必要はない |
| 公 表 | a. ソースプログラムを公表する b. ソースプログラムの公表は一定期間保留する(年 月 日まで) | |

§ 1. 概 要

1.1 目的

実変数ガンマ関数の逆数を倍精度で計算する。

1.2 計算方法

$$\Gamma(x+1) = x\Gamma(x)$$

を用いて変数の範囲を $2 \leq x < 3$ に変え、この範囲ではガンマ関数の多項式展開を用いる。

多項式展開の係数はWernerとCollingeの値を用いた。

[参考文献]

H. Werner and R. Collinge "Chebyshev Approximations to the Gamma Function"
Math. Comput. 15 195(1961)

§ 2. 使用法

2.1 呼び出し方法

ガンマ関数の逆数が必要なところに次のように書けばよい。

DRPGMA (X)

2.2 パラメータ

X: 倍精度実数型定数または変数名。

2.3 制限

Xのとりうる範囲に制限はないが、Xが絶対値の大きな負の数であると overflow を起こす。

($X > -50$ 位までは使える)

2.4 精度・使用時間

1回の呼び出しに約 0.5ms 位。精度は16~17桁程度。

§ 3. 備 考

- (1) 関数名 DRPGMA はこれを使うプログラム単位内で倍精度実数型の宣言を必要とする。
- (2) 内部でのエラー処理は一切行っていない。

No. 296 D6/QC/F/FFTCS

登録年月日 昭和46年6月28日

FAST FOURIER TRANSFORM (COMPLEX TRANSFORM)

高速フーリエ変換 (複素変換)

| | | |
|----------------|--|---------------------------------------|
| 作 成 | 作 成 者 久 原 由 美 子 | 作 成 年 月 日 昭和46年3月10日 |
| 形 式 | a. コンプリートプログラム | ⑥. サブルーチン d. 手続き |
| 使 用 言 語 | ①. FORTRAN d. PL/I | b. ALGOL e. その他 () c. FASP |
| 使 用 機 種 | FACOM 230-60 | |
| 使用メモリ数 | ①. コア(1.27)K語 | b. ディスクパック()K語 c. その他() |
| 使 用 機 器 構 成 | ①. カードリーダー d. 紙テープリーダー f. 磁気テープ() g. ディスクパック h. その他 () | ②. ラインプリンタ e. 紙テープパンチ c. カードパンチ |
| 利用者の義務 | a. プログラム名と作成者名を明記する | ② 明記する必要はない |
| 公 表 | ①. ソースプログラムを公表する b. ソースプログラムの公表は一定期間保留する(年 月 日まで) | |

§ 1. 概 要

1.1 目的

N個の時系列複素データX(0), X(1), ..., X(N-1)が与えられているとき、複素フーリエ変換は次式で表わすことができる。

$$C(j) = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} X(k) W^{jk}, \quad j = 0, 1, \dots, N-1 \quad \dots\dots\dots (1)$$

ここに $W = e^{2\pi i/N}$ である。 $\dots\dots\dots (2)$

(1)式の逆変換は

$$X(k) = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} C(j) W^{jk}, \quad k = 0, 1, \dots, N-1 \quad \dots\dots\dots (3)$$

と表わされる。

上式の形のままでC(j), j = 0, ..., N-1 を求めようとする、N²回の演算 (1回の

演算は1回の複素数の乗算と加算)を必要とし非常に時間がかかる。さらに4N個(2×2N(実数部N,虚数部N))の記憶場所を必要とする。この方法では、標本数が増加するにつれてその計算量は大変なものとなる。そこで最近CooleyとTukeyによって複素フーリエ変換の新しい算法が工夫されて、¹⁾飛躍的に計算量を減らすことができるようになった。これを一般に高速フーリエ変換(Fast Fourier Transform,略してFFT)と呼ばれている。FFTにもいろいろ工夫されたものがあるが、ここでは文献2)によっている。このプログラムは文献3)のALGOLの手続きをFACOM 230-60 FORTRANに書き換えたものである。

1.2 計算方法

N = 2^m のとき、(1)式のj, kを次のように表わす。

$$\left. \begin{aligned} k &= j_{m-1} \cdot 2^{m-1} + \dots + j \cdot 2 + j_0 \\ j &= k_{m-1} \cdot 2^{m-1} + \dots + k \cdot 2 + k_0 \end{aligned} \right\} \dots\dots\dots (4)$$

但し、j_{m-1}, ..., j₀, k_{m-1}, ..., k₀はそれぞれ0, 1をとる。即ちj, kの2進表現となっている。(以下normal orderと呼ぶ)。

これを用いると(1)式は

$$C(j_{m-1}, \dots, j_0) = \sum_{k_0} \sum_{k_1} \dots \sum_{k_{m-1}} X(k_{m-1}, \dots, k, k_0) W^{jk_{m-1} \cdot 2^{m-1} \dots j k_0} \dots (5)$$

となる。(2)と(4)式より

$$W^{jk_{m-1} \cdot 2^{m-1}} = W^{j_0 k_{m-1} \cdot 2^{m-1}} \dots\dots\dots (6)$$

したがって一番内側のk_{m-1}についての和はj₀, k_{m-2}, ..., k₀のみの関数となるから

$$X_1(j_0, k_{m-2}, \dots, k_0) = \sum_{k_{m-1}} X(k_{m-1}, \dots, k_0) W^{j_0 k_{m-1} \cdot 2^{\frac{m-1}{2}}} \dots\dots\dots (7)$$

となる。(7)式はk_{m-1} = 0, k_{m-1} = 1についての和であるから2N回の演算を必要とする。

同様に(2)と(4)式より

$$W^{jk_{m-1} \cdot 2^{\frac{m-1}{2}}} = W^{(j_{l-1} \cdot 2^{l-1} + \dots + j_0) k_{m-l} \cdot 2^{m-l}} \dots\dots\dots (8)$$

であるから

$$\begin{aligned} &X_l(j_0, \dots, j_{l-1}, k_{m-l-1}, \dots, k_0) \\ &= \sum_{k_{m-l}} X_{l-1}(j_0, \dots, j_{l-2}, k_{m-l}, \dots, k_0) W^{(j_{l-1} \cdot 2^{l-1} + \dots + j_0) k_{m-l} \cdot 2^{m-l}} \dots\dots\dots (9) \end{aligned}$$

この計算をlを1からmまでくり返し計算すると

$$C(j_{m-1}, \dots, j_0) = X_m(j_0, \dots, j_{m-1}) \dots\dots\dots (10)$$

が得られる。(10)式からわかるように最終結果は初めの2進表現の各ビットが逆順となっている(以下reverse binary orderと呼ぶ)。

すなわち

$$j = j_0 \cdot 2^{m-1} + \dots + j_{m-2} \cdot 2 + j_{m-1} \quad \dots \quad (11)$$

で表わされる。したがって最後に(4)の形に並べかえを行なう必要がある。

この場合の演算回数は

$$2Nm = 2N \log_2 N \quad \dots \quad (12)$$

となる。

(9)式において X_l を X_{l-2} で表現することもできる。すなわち k_{m-l} と k_{m-l-1} に対する和で表わす。以下に述べるサブルーチンはこの方法を用いている。

(3)式で表わされる逆変換の場合、(1)と(3)式をマトリックス表現で

$$\mathbf{C} = \frac{1}{\sqrt{N}} \mathbf{T} \mathbf{X}$$

$$\mathbf{X} = \frac{1}{\sqrt{N}} \mathbf{T}^* \mathbf{C}$$

と表わす。ただし、 \mathbf{T}^* は \mathbf{T} の複素共役行列である。

(14)式は

$$\mathbf{X} = \frac{1}{\sqrt{N}} (\mathbf{T} \mathbf{C}^*)^*$$

と表わすことができるから、最初に \mathbf{C} の複素共役ベクトルを作り、すでに述べた解法と同様の方法を用いる。次にその結果の複素共役ベクトルを作ることにより求まる。

[参考文献]

1) Cooley, J. W., Tukey, J. ~~W.~~^{W.}

An algorithm for the machine calculation of complex Fourier series.

Math. Comput. 19, 90, (Apr. 1965), 297-301.

2) Singleton, R. C.

On computing the fast Fourier Transform.

Comm. ACM 10 (Oct. 1967), 647-654.

3) Singleton, R. C.

ALGOL procedure for the Fast Fourier Transform.

Comm. ACM 11 (Nov. 1968), 773-776.

§ 2. 使用方法

2.1 呼び出し名

CALL FFTCS (C, N, M, INVERS, ILL)

2.2 パラメータ

C 単精度複素数型配列名。添字域の上限はN。
 入力としてN個の複素型データが与えられねばならない。出力として複素フーリエ変換の結果が与えられる。

N } 整数型変数
 M } N = 2 * * M

INVERS 論理型変数 .TRUE. または .FALSE. を与える。
 INVERSが .TRUE. のとき (1)式が計算される。
 INVERSが .FALSE. のとき (3)式が計算される。

ILL 整数型変数名。
 0……入力パラメータが正常なときにこの値が与えられる。
 1……入力パラメータに誤りがあり演算できないときに与えられる。
 即ち N≠ 2 * * Mのとき、またはM≥20のとき。

2.3 使用しているサブルーチン及び組込み関数

FFT2, REORDER
 CMLPX

§ 3. 備 考

FFTS (富士通提供) との計算時間及び使用語数の比較

テストデータとして、実部 = $f(t) = \begin{cases} 1 & 0 < t < \pi \\ 0 & \pi < t < 2\pi \end{cases}$, 虚部 = 0 なる関数を用いた。

| N | | FFTCS | FFTS |
|--------------|-------|-------------------|-------------------|
| 3 2 | | 2 9 ^{ms} | 6 0 ^{ms} |
| 6 4 | | 6 7 | 1 6 5 |
| 1 2 8 | | 1 5 9 | 3 2 7 |
| 2 5 6 | | 3 4 9 | 8 5 3 |
| 5 1 2 | | 7 8 0 | 1 7 3 9 |
| 1 0 2 4 | | 1 6 3 0 | 3 7 8 3 |
| 使用 語 数 | プログラム | 1 2 7 0 | 3 9 4 |
| | 変 数 | 2 N | 4 N |

註) FFTCSとFFTSとでは配列に与えられる結果の順序が異なる。

No. 297 D6/QC/F/FFTRS

登録年月日 昭和46年6月28日

FAST FOURIER TRANSFORM (REAL DATA TRANSFORM)

高速フーリエ変換 (実数データの変換)

| | | |
|--------|---|---|
| 作成 | 作成者 久原由美子 | 作成年月日 昭和46年3月10日 |
| 形式 | a. コンピュータプログラム | ④. サブルーチン d. 手続き |
| 使用言語 | ①. FORTRAN d. PL/I | b. ALGOL e. その他 () c. 関数 e. 関数手続き |
| 使用機種 | FACOM 230-60 | |
| 使用メモリ数 | ①. コア(2.4)K語 | b. ディスクパック()K語 c. その他() |
| 使用機器構成 | ①. カードリーダー d. 紙テープリーダー f. 磁気テープ()ユニット g. ディスクバック h. その他() | ②. ラインプリンタ e. 紙テープパンチ c. カードパンチ |
| 利用者の義務 | a. プログラム名と作成者名を明記する | ③. 明記する必要はない |
| 公表 | ①. ソースプログラムを公表する b. ソースプログラムの公表は一定期間保留する(年 月 日まで) | |

§ 1. 概要

1.1 目的

COMPLEX TRANSFORM を参照。

2個の実数データに対し、フーリエ cosine, sine 係数

$$a_k = \frac{1}{N} \sum_{j=0}^{2N-1} X_j \cos(\pi j k / N), \quad k = 0, 1, \dots, N \quad \dots\dots (1)$$

$$b_k = \frac{1}{N} \sum_{j=0}^{2N-1} X_j \sin(\pi j k / N), \quad k = 0, 1, \dots, N$$

を計算する。または、N個の cosine 係数、N個の sine 係数が与えられているとき

$$X_j = \frac{a_0}{2} + \sum_{k=1}^{N-1} [a_k \cos(\pi j k / N) + b_k \sin(\pi j k / N)] + \frac{a_N}{2} \cos(\pi j), \quad \dots\dots (2)$$

$$j = 0, 2, \dots, 2N-1$$

を計算する。

1.2 計算方法

2N個の実数データが $X(0), \dots, X(2N-1)$ に与えられているとき、虚数部を0として複素フーリエ変換を行なう。その後で実数データの場合、複素フーリエ係数 $C(j), j=1, 2, \dots, 2N-1$ の間に

$$C(2N-j) = C^*(j), \quad j=1, 2, \dots, N-1 \quad (\text{但し } * \text{ は複素共役を表わす})$$

が成り立つことを用いて、cosine 係数

$$A(j) = \begin{cases} C(0) + C^*(0) \\ C(j) + C^*(2N-j), \quad j=1, 2, \dots, N \end{cases}$$

sine 係数

$$B(j) = \begin{cases} i(C^*(0) - C(0)) \\ i(C^*(2N-j) - C(j)), \quad j=1, 2, \dots, N \end{cases}$$

を求める。逆の場合はこの逆をたどって変換する。

[参考文献]

No. 296 D6/QC/F/FFTCSの項を参照

§ 2. 使用方法

2.1 呼び出し名

CALL FFTRS (C, NN, M, INVERS, ILL)

2.2 パラメータ

C 単精度複素数型配列名。添字域の上限はNN。

INVERSが.TRUE. のとき、入力として最初のNN個のデータがCの実部に、残りのNN個がCの虚部に与えられねばならない。出力として最初のN個の結果がCの実部に、残りのN個の結果がCの虚部に与えられる。

INVERSが.FALSE. のとき、入力として最初のN個のデータがCの実部に、残りのN個のデータがCの虚部に与えられねばならない。出力として最初のNN個の結果がCの実部に、残りのNN個の結果がCの虚部に与えられる。

NN } 整数型変数。

M } $N = 2 * * M, NN = N + 1$

INVERS 論理型変数。TRUE. または.FALSE. を与える。

INVERSが.FALSE. のとき(1)式が計算される。結果は a_k がCの実部に、 b_k

がCの虚部に与えられる。

INVERSが.TRUE. のとき(2)式が計算される。

結果は最初のN個がCの実部に、残りのN個がCの虚部に与えられる。

ILL 整数型変数名。

0……入力パラメータが正常なときにこの値が与えられる。

1……入力パラメータに誤りがあり、演算できないときに与えられる。即ち

$NN \neq 2 * * M + 1$ のとき、または $M \geq 20$ のとき。

2.3 使用しているサブルーチン、組込み関数及び基本外部関数

FFT 2, REVFFT, RTRAN, REORDER

CMPLX

SIN

§ 3. 備 考

計算時間及び使用語数

テストデータとして 矩形波 $f(t) = \begin{cases} 1 & 0 < t < \pi \\ 0 & \pi < t < 2\pi \end{cases}$ を用いた。

| N | | FFTRS |
|--------------|-------|-----------|
| 3 2 | | 3 4 ms. |
| 6 4 | | 7 7 |
| 1 2 8 | | 1 7 0 |
| 2 5 6 | | 3 6 2 |
| 使用 語 数 | プログラム | 2 4 0 0 |
| | 変 数 | 2 (N + 1) |

No. 298 C3/QU/F/GAMMAD

登録年月日 昭和46年6月28日

GAMMA FUNCTION

ガンマ関数

| | | |
|-------------|---|--|
| 作 成 | 作 成 者 田 村 英 之 / | 作成年月日 昭和46年4月1日 |
| 形 式 | a. コンプリートプログラム | ⑥. サブルーチン d. 手続き c. 関数 e. 関数手続き |
| 使用言語 | ①. FORTRAN d. PL/I | b. ALGOL e. その他() c. FASP |
| 使用機種 | FACOM 230-60 | |
| 使用メモリ数 | ①. コア(0.7)K語 b. ディスクパック()K語 c. その他() | |
| 使用機器 構 成 | ①. カードリーダー d. 紙テープリーダー f. 磁気テープ()ユニット g. ディスクパック h. その他() | ②. ラインプリンタ e. 紙テープパンチ c. カードパンチ |
| 利用者の義務 | a. プログラム名と作成者名を明記する ② 明記する必要はない | |
| 公 表 | ①. ソースプログラムを公表する b. ソースプログラムの公表は一定期間保留する(年 月 日まで) | |

§ 1. 概 要

1.1 目的

ガンマ関数： $\Gamma(z) = \int_0^{\infty} e^{-t} t^{z-1} dt$ (ただし z は実数)

を、ベキ級数、漸近展開の最適併用条件により、誤差と計算時間を最小にする条件で計算する。

〔参考文献〕

森口ほか： 数学公式Ⅲ， P. 1 (岩波全書)

田村： 九大工学集報， 第43巻 第4号， P. 556 (昭45. 6)

§ 2. 使用法

2.1 呼び出し方法

CALL GAMMAD (Z, G, ILL)

2.2 パラメータ

Z…………… $\Gamma(z)$ のZを与える。倍精度実数型変数名または倍精度実定数。

G……………結果の $\Gamma(z)$ がセットされる倍精度実数型変数名。

ILL……………整数型変数名。サブルーチンからもどったときの状態がセットされる。

0……………正常に解が求まったとき。

1…………… $\Gamma(Z)$ が極であるときセットされる。

$|Z| < 10^{-50}$ および $|Z - (\text{負整数})| / |Z| < 10^{-16}$ のとき。

2.3 制限

$|Z| < 10^{-50}$ および $|Z - (\text{負整数})| / |Z| < 10^{-16}$ は計算不可能 ($\pm\infty$, 1位の極)。

この場合の結果は ILL = 1, G = 10^{50} 。

2.4 使用ルーチン

組込み関数 DABS, DSQRT, DLOG, DEXP

2.5 所要時間

Z = 1 ~ 2 に対して一回の呼び出しで 約10msec. $|Z|$ の増大とともにやゝ大となる。

2.6 精度

相対誤差 $10^{-16} \times (1 + |Z - 1.5|)$ 以下。Z = 1 ~ 2 では約 10^{-17} 程度。

2.7 備考

プログラムの一部を変更することにより、使用する計算機の語長に応じて、誤差と計算時間を最小にする条件で計算できる。語長 (1語のビット数) が与えられると相対誤差の下限はおおよそ $2^{-(\text{仮数ビット数}-1)}$ (相対誤差の上限はこの値より小さくならない) で、このプログラムによる $\Gamma(z)$ ($z = 1 \sim 2$) の相対誤差は、この値よりやゝ大となる。

No. 299 C3/QU/F/PRESNL

登録年月日 昭和46年6月28日

Generalized Fresnel Integral

一般化されたフレネル積分

| | | |
|----------------|--|--|
| 作 成 | 作 成 者 田 村 英 之 | 作 成 年 月 日 昭 和 46 年 4 月 28 日 |
| 形 式 | a. コンプリートプログラム | ⑤. サブルーチン d. 手続き c. 関数 e. 関数手続き |
| 使 用 言 語 | ④. FORTRAN d. PL/I | b. ALGOL e. その他 () c. FASP |
| 使 用 機 種 | FACOM 230-60 | |
| 使 用 メ モ リ 数 | ④. コア () K語 | b. ディスクパック () K語 c. その他 () |
| 使 用 機 器 構 成 | ④. カードリーダー d. 紙テープリーダー f. 磁気テープ () ユニット g. ディスクパック h. その他 () | ⑤. ラインプリンタ e. 紙テープパンチ c. カードパンチ |
| 利 用 者 の 義 務 | a. プログラム名と作成者名を明記する | ⑤ 明記する必要はない |
| 公 表 | ④. ソースプログラムを公表する b. ソースプログラムの公表は一定期間保留する (年 月 日まで) | |

§ 1. 概 要

1.1 目的

積分: $\int_0^x \exp(jt^p) dt = C_p(x) + jS_p(x)$, $j = \sqrt{-1}$, $p = 1 \sim \infty$

を(かりに)一般化されたフレネル積分と呼ぶ($p=2$ が通常のフレネル積分であるが、定義の仕方で変数と積分値は定数倍だけ異なる)。

この計算を、ベキ級数、漸近展開の最適併用条件により計算する。

1.2 計算方法

$x \rightarrow 0$ ではベキ級数を、 $x \rightarrow \infty$ では漸近展開を使い、両方法の誤差が等しくなる x で切りかえを行なう。詳細は下記文献を参照してください。

[参考文献]

森口ほか: 数学公式III, P. 22, 261 (岩波全書)

田村: 九大工学集報, vol. 43 No. 4, P. 550 (昭45. 6)

富士通: FACOM 230-60 FORTRAN 解説編(II), (EX-061-3-4), P. 134

§ 2. 使用法

2.1 呼び出し方法

CALL PRESNL (P, X, C, S)

2.2 パラメータ

P……指数 p ($1 \sim \infty$) を与える倍精度実数型変数名または倍精度実定数。

X……独立変数 x (> 0) を与える倍精度実数型変数名または倍精度実定数。

C……結果の積分値の実数部がセットされる。倍精度実数型変数名。

S……結果の積分値の虚数部がセットされる。倍精度実数型変数名。

2.3 制限

$$1 > P < \infty, X \geq 0$$

2.4 使用ルーチン

組込み関数: DABS, DLOG, DCOS, DSIN, DSQRT

2.5 所要時間

1回の呼び出しで約20msec.

2.6 精度

P, Xによって異なるが、FACOM 230-60倍精度の場合、 10^{-10} 以上が保証される。

20で最も誤差が大であるが、 $x^p \geq 32$ 、 $x^p \geq 9$ では 10^{-15} 程度であろう。

2.7 備考

(i) プログラムの一部を変更することにより、計算機の語長に応じて精度最良の条件で計算ができる。

(ii) 通常フレネル積分の定義: $\int_0^x \exp(j \frac{2}{\pi} t^2) dt = C(x) + jS(x)$

による $C(x)$ 、 $S(x)$ はそれぞれ、

$$C(x) = \sqrt{\frac{2}{\pi}} C_2(\sqrt{\frac{\pi}{2}} x), \quad S(x) = \sqrt{\frac{2}{\pi}} S_2(\sqrt{\frac{\pi}{2}} x)$$

である。

また、富士通ライブラリ FRESL (X, C, S, ILL) の定義

$$\int_0^x \frac{e^{-jt}}{\sqrt{2\pi t}} dt = C(x) - jS(x)$$

による $C(x)$ 、 $S(x)$ との関係は

$$C(x) = C_2(\sqrt{2x/\pi}), \quad S(x) = S_2(\sqrt{2x/\pi})$$

である。

No. 300 D1/QU/F/CGJQ

登録年月日 昭和46年6月28日

Coefficients of Gauss-Jacobi Quadrature

Gauss-Jacobi積分公式の係数

| | | | | |
|-------------|---|-----------------------------------|--------------------|--|
| 作 成 | 作 成 者 塩 川 浩 三 | | 作成年月日 昭和46年5月1日 | |
| 形 式 | a. コンプリートプログラム | ⑥. サブルーチン d. 手続き | c. 関数 e. 関数手続き | |
| 使用言語 | ①. FORTRAN d. PL/I | b. ALGOL e. その他 () | c. FASP | |
| 使用機種 | FACOM 230-60 | | | |
| 使用メモリ数 | ①. コア(1)K語 | b. ディスクパック()K語 | c. その他() | |
| 使用機器 構 成 | ①. カードリーダー d. 紙テープリーダー f. 磁気テープ() g. ディスクパック h. その他() | ②. ラインプリンタ e. 紙テープパンチ)ユニット | c. カードパンチ | |
| 利用者の義務 | a. プログラム名と作成者名を明記する | | ③ 明記する必要はない | |
| 公 表 | ④. ソースプログラムを公表する b. ソースプログラムの公表は一定期間保留する(年 月 日まで) | | | |

§ 1. 概要

1.1 目的

$(1-x)^r(1+x)^s$ を重み関数とする Gauss-Jacobi 積分公式 (1)

$$\int_{-1}^1 (1-x)^r (1+x)^s f(x) dx = 2^{r+s+1} B(r+1, s+1) \sum_{k=1}^n H_k f(a_k)$$

の分点 a_k 、重み H_k を求める。

(1)で $B(\alpha, \beta)$ はベータ関数である。

1.2 計算法

分点 a_k は n 次の Jacobi polynomial $G_n^{r,s}(x)$ の零点であることを利用し、適当な近似値を Newton の方法により修正して求める。

重み H_k は次式で与えられる。

$$H_k = \frac{4^{n-1} B(r+n, s+n) B(r+s+n, n)(r+s+2n-1)}{B(r+1, s+1) G_{n-1}^{r,s}(a_k) \frac{d}{dx} (G_n^{r,s} a_k)}$$

〔参考文献〕

A. H. Stroud and D. Secrest

“Gaussian Quadrature Formulas” Prentice-Hall, Inc.

§ 2. 使用法

2.1 呼び出し方法

CALL CGJQ (N, R, S, A, H, B, C, O, ILL)

2.2 パラメータ

N 整数型定数または変数名。分点の数を与える。

R 倍精度実数型定数または変数名。

(1)式における重み関数の $(1-x)$ のべきを与える。 $R \geq -0.999$ でなければならない。

S 倍精度実数型定数または変数名。

(1)式における重み関数の $(1+x)$ のべきを与える。 $S \geq -0.999$ でなければならない。

A 倍精度実数型配列名。大きさN以上の一次元配列。

分点が値の大きい方から順にセットされて戻る。 $A(K) > A(L)$ ($K > L$)

H 倍精度実数型配列名。大きさN以上の一次元配列。

分点に対応する重みがセットされて戻る。

B, C 倍精度実数型配列名。いずれも大きさN以上の一次元配列。作業領域として用いる。

O 実数型定数または変数名。分点の収束判定値を与える。

 $O \leq 0.0$ であれば 10^{-16} が与えられたものとして計算する。

ILL 整数型変数名。

このサブルーチンが呼ばれる時： 分点の近似値の選び方を指示するパラメータ。(注)

戻ってきた時： エラー処理のパラメータ。(エラー処理参照)

ILL=0 正しく処理されている

ILL>0 異常

注) このサブルーチンを呼ぶとき、 $ILL < 0$ であれば分点の近似値が $A(K)$ ($K=1, 2, \dots, N$)

に与えられているものとして計算を行なう。

2.3 制限

$N \geq 1$, $R \geq -0.999$, $S \geq -0.999$ であれば計算を行なうが、Nがあまり大きいと overflow を起す。たとえば $N=100$, $R=-0.998$, $S=30$ で overflow が起きた。

2.4 エラー処理

次の場合はILLパラメータにそれぞれの値をセットして戻る。

a) $N \leq 0$ のとき ILL=1000

b) $R < -0.999$ または $S < -0.999$ のとき ILL=2000

c) a_k が40回反復修正しても収束しないとき ILL=R

d) $\sum_{k=1}^n A(K)$ が論理値 $\frac{N \cdot (S-R)}{2N+S+R}$ と異なるとき ILL=3000

いずれもエラーメッセージは出さない。

2.5 使用ルーチン

DABS

2.6 使用時間、精度

たとえば $R=S=0$, $N=10$ で64msec. 15桁以上の精度 ($O=0.0$ とした場合)。

§ 3. 備 考

R, Sがあまり大きいとき、Nが小さい場合、また大きすぎる場合に正しい解が得られないことが多い(この場合、ILLが3000になることが多い)。

次の場合、いずれもILL=3000となった。

$$R = -0.998, \quad S = 30, \quad N = 50$$

$$R = -0.998, \quad S = 50, \quad N = 2$$

$$R = -0.9, \quad S = 20, \quad N = 50$$

$$R = -0.9, \quad S = 50, \quad N = 2$$

No. 301 D1/QU/F/CGLQ

登録年月日 昭和46年6月28日

Coefficients of Gauss Laguerre Quadrature

Gauss Laguerre 積分公式の係数

| | | |
|-------------|--|---------------------------------------|
| 作 成 | 作 成 者 塩 川 浩 三 | 作成年月日 昭和46年5月1日 |
| 形 式 | a. コンプリートプログラム b. サブルーチン d. 手続き | c. 関数 e. 関数手続き |
| 使用言語 | Ⓐ. FORTRAN d. PL/I | b. ALGOL e. その他 () c. FASP |
| 使用機種 | FACOM 230-60 | |
| 使用メモリ数 | Ⓐ. コア(0.6)K語 | b. ディスクパック()K語 c. その他() |
| 使用機器 構 成 | Ⓐ. カードリーダー d. 紙テープリーダー f. 磁気テープ() g. ディスクパック h. その他 () | Ⓑ. ラインプリンタ e. 紙テープパンチ c. カードパンチ |
| 利用者の義務 | a. プログラム名と作成者名を明記する | Ⓑ 明記する必要はない |
| 公 表 | Ⓐ. ソースプログラムを公表する b. ソースプログラムの公表は一定期間保留する(年 月 日まで) | |

§ 1. 概 要

1.1 目的

$x^s e^{-x}$ を重み関数とする Gauss Laguerre 積分公式(1)

$$\int_0^{\infty} x^s e^{-x} f(x) dx \doteq \Gamma(s+1) \sum_{k=1}^n H_k f(a_k) \quad (1)$$

の分点 a_k と重み H_k ($k=1, 2, \dots, n$) を求める。

1.2 計算方法

分点 a_k は n 次の Laguerre Polynomial $L_n^s(x)$ の零点であることを利用し、適当な近似値を Newton の方法により修正して求める。重み H_k は次式を用いて求める。

$$H_k = \frac{\Gamma(n+s+1) \cdot a_k}{n! \Gamma(s+1) \{ (n+s) L_{n-1}^s(a_k) \}^2}$$

[参考文献]

A. H. Stroud and D. Secrest

"Gaussian Quadrature Formulas" Prentice-Hall, Inc.

§ 2. 使用法

2.1 呼び出し方法

CALL CGLQ (N, S, A, H, O, ILL)

2.2 パラメータ

- N 整数型定数または変数名。分点の数を与える。
- S 倍精度実数型定数または変数名。(1)式における重み関数のべきを与える。
 $S \geq -0.999$ でなければならない。
- A 倍精度実数型配列名。大きさN以上の一次元配列。
 分点が値の小さいものから順にセットされて戻る。 $A(K) < A(L) \quad (K < L)$
- H 倍精度実数型配列名。大きさN以上の一次元配列。
 分点に対応する重みがセットされて戻る。
- O 実数型定数または変数名。分点の収束判定値を入れておく。
 $0 \leq O$ であれば 10^{-16} が与えられたものとして計算する。
- ILL 整数型変数名。
 サブルーチンが呼ばれるとき、分点の近似値のえらび方を指示するパラメータを与える。
 戻ってくるときはエラー処理のパラメータをセットして戻る。
 $ILL = 0$ 正しく処理されている
 $ILL > 0$ エラーがあった(エラー処理参照)

注) このサブルーチンを呼ぶとき $1 \leq ILL < N$ であれば、分点 a_k ($k \geq ILL$) の近似値が $A(k)$ に与えられているものとして計算する。またILLが負であれば、すべての分点の近似値がAに与えられているものとして計算する。

2.3 制限

$N \geq 1$, $S \geq -0.999$ であれば計算を行なうが、Nがあまり大きいとoverflowを起こす。
 たとえば $S = -0.998$ のとき、 $N = 100$ ではoverflowが起きた。

2.4 エラー処理

次の場合はILL パラメータにそれぞれの値をセットして戻る。

- a) $N \leq 0$ のとき $ILL = 1000$
- b) $S < -0.999$ のとき $ILL = 2000$
- c) a_k が40回反復修正しても収束しないとき $ILL = k$
- d) $\sum_{k=1}^n A(K)$ が理論値 $N \cdot (N + S)$ と一致しないとき $ILL = 3000$

2.5 使用ルーチン

DABS

2.6 使用時間、精度

N, Oによって異なるが、たとえばS=0、N=10で55msec. 精度15桁以上。

§ 3. 備 考

(1) Sが大きいときは、Nが大きい場合またはNが小さすぎる場合に正しい解が得られない。

たとえば次の場合、ILL=3000となった。

$$S=20.0 \quad N=2, N=50$$

$$S=50.0 \quad N=5, N=40$$

$$(2) \int_0^{\infty} x e^{-x} f(x) dx = \frac{1}{2} \int_0^{\infty} t^{\alpha-1} e^{-t} f(t^2) dt$$

であるから、このサブルーチンを使って Gauss-Hermite 積分公式の分点、重みが計算できる。

No. 302 D1/QU/F/ROMBGS

登録年月日 昭和46年6月28日

No. 303 D1/QU/F/ROMBGD

Numerical Integration using Romberg's Algorithm

数値積分 Romberg 法

| | | |
|-------------|--|---------------------------------------|
| 作 成 | 作 成 者 塩 川 浩 三 | 作成年月日 昭和 46年 5月 1日 |
| 形 式 | a. コンプリートプログラム b. サブルーチン d. 手続き | ③. 関数 e. 関数手続き |
| 使用言語 | ①. FORTRAN d. PL/I | b. ALGOL e. その他 () c. FASP |
| 使用機種 | FACOM 230-60 | |
| 使用メモリ数 | ①. コア(0.4)K語 | b. ディスクパック()K語 c. その他() |
| 使用機器 構 成 | ①. カードリーダー d. 紙テープリーダー f. 磁気テープ()ユニット g. ディスクパック h. その他 () | ②. ラインプリンタ e. 紙テープパンチ c. カードパンチ |
| 利用者の義務 | a. プログラム名と作成者名を明記する | ② 明記する必要はない |
| 公 表 | ①. ソースプログラムを公表する b. ソースプログラムの公表は一定期間保留する(年 月 日まで) | |

§ 1. 概 要

1.1 目的

有限区間の定積分を計算する。

1.2 計算方法

Romberg法を用いる。

[参考文献]

H. Rutishauser

“Ausdehnung des Rombergshen Prinzips”

Numer. Math. 5 48 (1963)

§ 2. 使用法

2.1 呼び出し方法

関数副プログラムであるから積分値の必要な所に

ROMBGS (N, XI, XF, FUNC, T, EPS, ILL) (単精度)

または

ROMBGD (N, XI, XF, FUNC, T, EPS, ILL) (倍精度)

と書けばよい。

2.2 パラメータ

N 整数型定数または変数名。最大反復回数を与える。

XI 実数型定数または変数名。積分領域の下限を与える。

XF 実数型定数または変数名。積分領域の上限を与える。

FUNC 文関数または関数副プログラム名。被積分関数を与える。

FUNC (x) の形で定義されていること。

T 実数型配列名。大きさN以上の一次元配列。作業領域として用いる。

EPS 実数型変数名。相対誤差の収束判定値を与えておく。絶対誤差がセットされて戻る。

ILL 整数型変数名。

・ 0 以下の値を与えておく。

・ エラー処理のパラメータをセットして戻る。

ILL ≤ 0 正常

ILL > 0 異常

注)

①倍精度副プログラムROMBGDの場合、XI, XF, FUNC, T, EPS は倍精度実数型でなければならない。

②FUNCが関数副プログラムときは、EXTERNAL宣言が必要である。

③k 回目の反復で得られた積分値を I_k とすると

$|I_k - I_{k-1}| \leq I_k \cdot \text{EPS}$ で収束したとみなし、EPSに $|I_k - I_{k-1}|$ をセットして戻る。

2.3 エラー処理

次の場合はILLにそれぞれの値をセットして戻る。

| | |
|---------------------------------|------------|
| N < 0 のとき、ILLにN以上の値が与えられているとき | ILL=10000 |
| 2^{-k} (kは反復回数)がEPSより小さくなったとき | ILL=1000+k |
| N回反復しても収束しないとき | ILL=N+1 |

2.4 使用ルーチン

ROMBGS.....ABS

ROMBGD.....DABS

§ 3. 備 考

3.1 収束しないため ILL > 0 で戻った場合、次のように計算を続行できる。

```

DIMENSION T(10)
      :
      EPS=1.0E-6
      SEKIBN=ROMBGS ( 5 , 0.0 , FUNC , EPS , ILL ) ..... ( 1 )
      IF ( ILL . LE . 0 ) GO TO 100 ..... ( 2 )
      EPS=1.0E-6 ..... ( 2 )
      SEKIBN=ROMBGS ( 10 , 0.0 , FUNC , EPS , ILL ) ..... ( 3 )
100 .....
      .....
      .....
    
```

このとき(2)でEPSを大きくしてもよいが、ILLの値を与えてはいけない。

3.2 使用時間・精度

$\int \frac{1+x+x^3}{(1+x^2)^{3/2}} e^x dx$ を、xが0.5から0.5きざみで10まで変えていろいろな精度で計算した。

結果は以下の通りである。なお、精度は $\pm 2 \times \text{EPS}$ 程度である。

| EPS | ROMBGS | ROMBGD | SIMPD |
|------------|--------|--------|-------|
| 10^{-4} | 0.3 秒 | — 秒 | — 秒 |
| 10^{-6} | 0.07 | 1.0 | 2.3 |
| 10^{-12} | — | 6.9 | 75.0 |
| 10^{-15} | — | 13.9 | 389.0 |
| 10^{-17} | — | 18.8 | |

No. 304 Y3/QC/Z/DB01

登録年月日 昭和46年6月28日

CLEBSCH-GORDAN COEFFICIENT

クレブシュ・ゴルダン係数

No. 305 Y3/QC/Z/DB02

RACAH COEFFICIENT

ラカー係数

No. 306 Y3/QC/Z/DB03

NINE-J SYMBOL

9-J 係数

| | | | |
|------------|--|--------------------------|-------------------|
| 作成 | 作成者 田村太郎 | | |
| 改訂 | 改訂者1 | 板倉 守 | 改訂年月日 昭和44年4月23日 |
| | 改訂者2 | 上村 正康* | 昭和46年5月1日 |
| 形式 | a. コンプリートプログラム | b. サブルーチン d. 手続き | ③. 関数 e. 関数手続き |
| 使用言語 | ①. FORTRAN d. PL/I | b. ALGOL e. その他 () | c. FASP |
| 使用機種 | FACOM 230-60 | | |
| 使用メモリ数 | ①. コア $\begin{pmatrix} 1.3 \\ 1.4 \\ 1.0 \end{pmatrix}$ K語 | b. ディスクパック () K語 | c. その他 () |
| 使用機器 構成 | ①. カードリーダー d. 紙テープリーダー f. 磁気テープ () ユニット g. ディスクパック h. その他 () | ②. ラインプリンタ e. 紙テープパンチ | c. カードパンチ |
| | 利用者の義務 | | |
| 公表 | ①. プログラム名と作成者名を明記する | | b. 明記する必要はない |
| | ①. ソースプログラムを公表する b. ソースプログラムの公表は一定期間保留する (年 月 日まで) | | |

*登録されたプログラムは、上村氏が東大ライブラリを九大FACOM 230-60用書き換えられたものです。

§ 1. 概要

Clebsh-Gordan 係数、Racah 係数及び $9-j$ Symbol を計算する関数型サブプログラムである。これらのコードは出来るだけ手短かに、また理解し易いように組んだつもりである。

これらのコードは、Argument が非常に大きい場合（大体 200程度まで）に使えるようにしてある。従って、重い粒子による散乱・反応とか、High-energy projectilesによる反応、散乱の場合にも使用することができる。

§ 2. 使用法

2.0 使用方法

¥LIED, ¥DAFILE のマクロを用い、自分でLIEDの御御文を書いてください。

¥DAFILEのパラメータで指定するファイル名は、QU. RB. A. LIBです。

[例]

¥NO

¥QJOB

¥FORTRAN

FORTRAN ソースプログラム

¥LIED

NAME EXQTPRGM, ENTRY= (PRG. MAIN)

CALL SYSLIB

SGMT SEG 1

SELECT RELBIN, FDNAME 1

FIN

¥DAFILE FDNAME=FDNAME 1,

FLNAME=QU, RB. A. LIB

データ

¥JEND

なお、コントロールカードの詳細については「センターニュース No14」及びセンター受付プログラム相談室に備えてある「ファイル利用のための手引き」を参照ください。共用ボリューム、私用ファイルの使い方と同様です。

2.1 田村太郎氏作成のCLEB, RAC 7, NINEJは、Subroutine 型のSubprogram であったが、使用頻度、及び使用上の便利のために Function 型のSubprogramに書き直した。また、Argument が適当でない場合には、Error Message を書き出して、その値を 0.0 と出力するようにした。

これらのProgram は夫々Function Type であるから、計算式の途中にArray name と同じように使用出来、1 語長実数型の正しい値をそのまま使用することが出来る。

次にArgument の並べ方を説明する。

1) FUNCTION CLEB (FJ 1, FM 1, FJ 2, FM 2, FJ 3, FM 3)

このProgram は、Clebsch-Gordan 係数 (FJ 1, FM 1, FJ 2, FM 2, FJ 3, FM 3)

即 ($j_1, m_1, j_2, m_2, j_3, m_3$) の値を計算するものである。夫々のArgument の順序とTypeは

| Arguments | Type | Physical meaning |
|-----------|---------|------------------|
| F J 1 | 1 語長実数型 | j_1 |
| F M 1 | " | m_1 |
| F J 2 | " | j_2 |
| F M 2 | " | m_2 |
| F J 3 | " | j_3 |
| F M 3 | " | m_3 |

2) FUNCTION RAC 7 (FJ 1, FJ 2, FJ, FJ 3, EJ12, FJ23)

このProgram はRacah 係数W (FJ 1, FJ 2, FJ, FJ 3, FJ12, FJ23) 即 W ($j_1, j_2, J, j_3; j_{12}, j_{23}$) の値を計算するものである。夫々のArguments の順序とTypeは

| Arguments | Type | Physical meaning |
|-----------|---------|------------------|
| F J 1 | 1 語長実数型 | j_1 |
| F J 2 | " | j_2 |
| F J | " | J |
| F J 3 | " | j_3 |
| F J 12 | " | j_{12} |
| F J 23 | " | j_{23} |

3) FUNCTION U9J (FJ 1, FJ 2, FJ 3, FJ 4, FJ 5, FJ 6, FJ 7, FJ 8, FJ 9)

このProgramは9J係数

$$\left\{ \begin{array}{l} \text{FJ 1, FJ 2, FJ 5} \\ \text{FJ 3, FJ 4, FJ 6} \\ \text{FJ 7, FJ 8, FJ 9} \end{array} \right\} \text{即} \left\{ \begin{array}{l} j_1, j_2, j_5 \\ j_3, j_4, j_6 \\ j_7, j_8, j_9 \end{array} \right\} \equiv \left\{ \begin{array}{l} j_1, j_2, j_{12} \\ j_3, j_4, j_{34} \\ j_{13}, j_{24}, J \end{array} \right\}$$

の値を計算するものである。それぞれArgumentの順序とTypeは

| Arguments | Type | Physical meaning |
|-----------|---------|------------------|
| F J 1 | 1 語長実数型 | j_1 |
| F J 2 | " | j_2 |
| F J 3 | " | j_3 |
| F J 4 | " | j_4 |
| F J 5 | " | j_{12} |
| F J 6 | " | j_{34} |
| F J 7 | " | j_{13} |
| F J 8 | " | j_{24} |
| F J 9 | " | J |

2.2 エラーメッセージ

これらのProgramはArgumentの入れ方が適当でないときに次のError Messageを書き出し、出力を0.0としてRETURNする。

I) FUNCTION CLEB (FJ 1, FM 1, FJ 2, FM 2, FJ 3, FM 3)

(イ) ERROR * THERE ARE SOME NEGATIVE A. M. IN CLEB * J 1 = J 2 = J 3 =

j_1, j_2, j_3 のどれかの値に負の実数が入っている場合

(ロ) ERROR * SUM OF MAGNETIC QUANTUM NUMBER IS NOT ZERO IN CLEB * M 1 = M 2 = M 3 =

$m_1 + m_2 - m_3$ の値が0にならない場合

(ハ) ERROR * SUM OF ANGULAR MOMENTUM IS NOT INTEGER IN CLEB * J 1 = J 2 = J 3 =

$j_1 + j_2 + j_3$ が整数にならない場合

(ニ) ERROR * TRIANGULAR CONDITION IS VIOLATED IN CLEB * J1=...

J2=..... J3=.....

j_1, j_2, j_3 が三角条件を満たさない場合

(ホ) ERROR * ABS (M. Q. N). GT. A.M IN CLEB * J1=....., M1=....., J2=...

M2=....., J3=....., M3=.....

$m_i > j_i$ なる Arguments が入っていた場合

(ヘ) ERROR * INVALID USE OF (J.M) PAIRS IN CLEB * J1=....., M1=.....,

J2=....., M2=....., J3=....., M3=.....

$(j_i + m_i)$ が整数にならないものがある場合

II) FUNCTION RAC7 (FJ1, FJ2, FJ, FJ3, FJ12, FJ23)

(イ) ERROR * SOME ARGUMENTS ARE NEGATIVE IN RAC7 * J1=.....,

J2=....., J=....., J3=....., J12=....., J23=.....

Arguments のどれかに負の実数が入っている場合

(ロ) ERROR * INVARID TRIANGULAR CONDITION IN RAC7 * J1=....., J2=

....., J=....., J3=....., J12=....., J23=.....

Arguments の組に三角条件を満たさないものがある。

III) FUNCTION U9J (FJ1, FJ2, FJ3, FJ4, FJ5, FJ6, FJ7, FJ8, FJ9)

(イ) ERROR * SOME ARGUMENTS ARE NEGATIVE IN UNINJ ***** J1=.....,

J2=....., J3=....., J4=....., J5=....., J6=....., J7=....., J8=....., J9=.....

Arguments のどれかに負の数が入っている場合

(ロ) ERROR * TRIANGULAR CONDITION IS INVALID IN UNINEJ ***** J1=

....., J2=....., J3=....., J4=....., J5=....., J6=....., J7=....., J8=.....,

J9=.....

Arguments の組に三角条件を満たさないものがある。

2.3 所要時間：Arguments の大きさに依るが Test に使用した程度の Argument について

\$ CLEB で平均約1/40秒

\$ RAC7 で 1/20秒

\$ U9J で 1/10秒

程度かかるものと思われたい。

2.4 精度：有効数字6桁以上

2.5 制限事項

これらの Program は核反応の計算に使用されることを念頭においてあるので Arguments の最大値は 200 近くまで使用出来るが、Argument の値が大きい場合多少精度は落ちるようである。テストの項参照。

2.6 使用ルーチン

U9J を使用する場合には RAC 7 を同時に使用すること。RAC 7, CLEB は夫々単一、独立のルーチンである。

2.7 備 考

〈注1〉“CLEB, RAC 7 の計算には、 $\log_e(n!)$ の和の計算が分子・分母のそれぞれに入るので、Arguments が大きすぎる場合には、Overflow する危険がある”と田村氏は注意しているが、普通の使い方をする場合にはほとんどその危険性はないものと思われる。

〈注2〉CLEB と RAC 7 ではそれぞれ $\log_e(n!)$ を計算を始める前につくっているが、ここで HITAC 5020 の計算機では、JOB の最初には Initial Clearance を行なうが、JOB の途中では Clear しない特性を生かして FACLOG (3) の値が計算されたか、されていないかを判定して計算されていない場合には、FACLOG ($\log_e n!$) をつくる。

即、最初に CLEB なり RAC 7 なりを呼んだ時にだけこれらの値を計算して Store しておくようにしてある。従って Initial Clearance を行わない計算や Subprogram から Return する度に Clearance を行う計算機では、これらの値を MAIN Routine で計算しておいて Common で受け渡すようにすればよい。

参考文献

1. M. E. Rose, “Elementary Theory of Angular Momentum”
2. A. R. Edmond, “Angular Momentum in Quantum Mechanics”
3. A. De-Shalit and I. Talmi, “Nuclear Shell Theory”
4. T. Inoue, “Table of the Clebsch-Gordan Coefficients”
5. T. Ishidzu et al, “Table of the Racah Coefficients”
6. K. Smith and J. W. Stevenson, “A Table of Wigner 9j Coefficients for integral and half integral values of the parameters”, ANL-5776, and 5886

§ 3. 使用例

$$1) (j_1, m_1, j_2, m_2, | j_3, m_3) = (2, 1, 3, -2 | 4, -1)$$

の値が必要な場合には、この値の必要な箇所に於て

$$\text{CLEBSH} = \text{CLEB}(2.0, 1.0, 3.0, -2.0, 4.0, -1.0)$$

とにおいてCLEBを呼べばCLEBSHには0.59160817の値が返ってくる。

$$2) W(a, b, c, d; e, f) = W(2, 3, 3/2, 5/2; 3, 1/2)$$

の値が必要な場合には、この値の必要な箇所に於て

$$\text{RACAH} = \text{RAC 7}(2.0, 3.0, 1.5, 2.5, 3.0, 0.5)$$

とにおいてRAC 7を呼べば

$$\text{RACAH} = -0.4638501 \text{ の値が返ってくる。}$$

$$3) P = (2, 1, 3, -2 | 4, -1) \cdot W(2, 3, 3/2, 5/2; 3, 1/2)$$

の値が必要な場合、この値の必要な箇所に於て

$$P = \text{CLEB}(2.0, 1.0, 3.0, -2.0, 4.0, -1.0) * \\ \text{RAC 7}(2.0, 3.0, 1.5, 2.5, 3.0, 0.5)$$

とにおいてCLEBとRAC 7を呼べばPには-0.08660257の値が返ってくる。

$$4) Q = \begin{Bmatrix} 1/2 & 2 & 3/2 \\ 3/2 & 3 & 5/2 \\ 2 & 3 & 4 \end{Bmatrix} \cdot \begin{Bmatrix} 1/2 & 2 & 3/2 \\ 3/2 & 3 & 5/2 \\ 2 & 3 & 3 \end{Bmatrix}$$

の値が必要な場合には、この値の必要な箇所に於て

$$Q = \text{U 9 J}(0.5, 2.0, 1.5, 3.0, 1.5, 2.5, 2.0, 3.0, 4.0) * \\ \text{U 9 J}(0.5, 2.0, 1.5, 3.0, 1.5, 2.5, 2.0, 3.0, 3.0)$$

とにおいてU 9 Jを呼べばQには-0.00001283の値が返ってくる。

$$5) R = W(2, 3, 3/2, 5/2; 3, 1/2) \cdot \begin{Bmatrix} 1/2 & 2 & 3/2 \\ 3/2 & 3 & 5/2 \\ 2 & 3 & 4 \end{Bmatrix}$$

の値が必要な場合にはこの値の必要な箇所に於て

$$R = \text{RAC 7}(2.0, 3.0, 1.5, 2.5, 3.0, 0.5) * \\ \text{U 9 J}(0.5, 2.0, 1.5, 3.0, 1.5, 2.5, 2.0, 3.0, 4.0)$$

とにおいてRAC 7とU 9 Jを呼べばRは0.00058451の値が返ってくる。