NAG Library Routine Document

S21BCF

Note: before using this routine, please read the Users' Note for your implementation to check the interpretation of ***bold italicised*** terms and other implementation-dependent details.

1 Purpose

S21BCF returns a value of the symmetrised elliptic integral of the second kind, via the function name.

2 Specification

```
FUNCTION S21BCF (X, Y, Z, IFAIL)
REAL (KIND=nag_wp) S21BCF

INTEGER            IFAIL
REAL (KIND=nag_wp) X, Y, Z
```

3 Description

S21BCF calculates an approximate value for the integral

$$R_D(x,y,z) = \tfrac{3}{2}\int_0^\infty \frac{dt}{\sqrt{(t+x)(t+y)(t+z)^3}}$$

where x, $y \geq 0$, at most one of x and y is zero, and $z > 0$.

The basic algorithm, which is due to Carlson (1979) and Carlson (1988), is to reduce the arguments recursively towards their mean by the rule:

$$\begin{array}{lcl}
x_0 & = & x, y_0 = y, z_0 = z \\
\mu_n & = & (x_n + y_n + 3z_n)/5 \\
X_n & = & (1 - x_n)/\mu_n \\
Y_n & = & (1 - y_n)/\mu_n \\
Z_n & = & (1 - z_n)/\mu_n \\
\lambda_n & = & \sqrt{x_n y_n} + \sqrt{y_n z_n} + \sqrt{z_n x_n} \\
x_{n+1} & = & (x_n + \lambda_n)/4 \\
y_{n+1} & = & (y_n + \lambda_n)/4 \\
z_{n+1} & = & (z_n + \lambda_n)/4
\end{array}$$

For n sufficiently large,

$$\epsilon_n = \max(|X_n|, |Y_n|, |Z_n|) \sim \left(\tfrac{1}{4}\right)^n$$

and the function may be approximated adequately by a fifth order power series

$$R_D(x,y,z) = \quad 3\sum_{m=0}^{n-1} \tfrac{4^{-m}}{(z_m+\lambda_n)\sqrt{z_m}}$$

$$+\frac{4^{-n}}{\sqrt{\mu_n^3}}\left[1 + \frac{3}{7}S_n^{(2)} + \frac{1}{3}S_n^{(3)} + \frac{3}{22}\left(S_n^{(2)}\right)^2 + \frac{3}{11}S_n^{(4)} + \frac{3}{13}S_n^{(2)}S_n^{(3)} + \frac{3}{13}S_n^{(5)}\right]$$

where $S_n^{(m)} = \left(X_n^m + Y_n^m + 3Z_n^m\right)/2m$. The truncation error in this expansion is bounded by $\dfrac{3\epsilon_n^6}{\sqrt{(1-\epsilon_n)^3}}$ and the recursive process is terminated when this quantity is negligible compared with the ***machine precision***.

The routine may fail either because it has been called with arguments outside the domain of definition, or with arguments so extreme that there is an unavoidable danger of setting underflow or overflow.

Note: $R_D(x, x, x) = x^{-3/2}$, so there exists a region of extreme arguments for which the function value is not representable.

4 References

Abramowitz M and Stegun I A (1972) *Handbook of Mathematical Functions* (3rd Edition) Dover Publications

Carlson B C (1979) Computing elliptic integrals by duplication *Numerische Mathematik* **33** 1–16

Carlson B C (1988) A table of elliptic integrals of the third kind *Math. Comput.* **51** 267–280

5 Arguments

1: X – REAL (KIND=nag_wp) *Input*

2: Y – REAL (KIND=nag_wp) *Input*

3: Z – REAL (KIND=nag_wp) *Input*

On entry: the arguments x, y and z of the function.

Constraint: X, Y ≥ 0.0, Z > 0.0 and only one of X and Y may be zero.

4: IFAIL – INTEGER *Input/Output*

On entry: IFAIL must be set to 0, -1 or 1. If you are unfamiliar with this argument you should refer to Section 3.4 in How to Use the NAG Library and its Documentation for details.

For environments where it might be inappropriate to halt program execution when an error is detected, the value -1 or 1 is recommended. If the output of error messages is undesirable, then the value 1 is recommended. Otherwise, if you are not familiar with this argument, the recommended value is 0. **When the value -1 or 1 is used it is essential to test the value of IFAIL on exit.**

On exit: IFAIL $= 0$ unless the routine detects an error or a warning has been flagged (see Section 6).

6 Error Indicators and Warnings

If on entry IFAIL $= 0$ or -1, explanatory error messages are output on the current error message unit (as defined by X04AAF).

Errors or warnings detected by the routine:

IFAIL $= 1$

On entry, either X or Y is negative, or both X and Y are zero; the function is undefined.

IFAIL $= 2$

On entry, Z ≤ 0.0; the function is undefined.

IFAIL $= 3$

On entry, either Z is too close to zero or both X and Y are too close to zero: there is a danger of setting overflow. See also the Users' Note for your implementation.

IFAIL $= 4$

On entry, at least one of X, Y and Z is too large: there is a danger of setting underflow. On softfailure the routine returns zero. See also the Users' Note for your implementation.

IFAIL = −99

An unexpected error has been triggered by this routine. Please contact NAG.

See Section 3.9 in How to Use the NAG Library and its Documentation for further information.

IFAIL = −399

Your licence key may have expired or may not have been installed correctly.

See Section 3.8 in How to Use the NAG Library and its Documentation for further information.

IFAIL = −999

Dynamic memory allocation failed.

See Section 3.7 in How to Use the NAG Library and its Documentation for further information.

7 Accuracy

In principle the routine is capable of producing full ***machine precision***. However round-off errors in internal arithmetic will result in slight loss of accuracy. This loss should never be excessive as the algorithm does not involve any significant amplification of round-off error. It is reasonable to assume that the result is accurate to within a small multiple of the ***machine precision***.

8 Parallelism and Performance

S21BCF is not threaded in any implementation.

9 Further Comments

You should consult the S Chapter Introduction which shows the relationship of this function to the classical definitions of the elliptic integrals.

10 Example

This example simply generates a small set of nonextreme arguments which are used with the routine to produce the table of low accuracy results.

10.1 Program Text

```
    Program s21bcfe

!     S21BCF Example Program Text

!     Mark 26 Release. NAG Copyright 2016.

!     .. Use Statements ..
      Use nag_library, Only: nag_wp, s21bcf
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter               :: nout = 6
!     .. Local Scalars ..
      Real (Kind=nag_wp)               :: rd, x, y, z
      Integer                          :: ifail, ix, iy
!     .. Intrinsic Procedures ..
      Intrinsic                        :: real
!     .. Executable Statements ..
      Write (nout,*) 'S21BCF Example Program Results'

      Write (nout,*)
      Write (nout,*) '    X      Y      Z        S21BCF'
      Write (nout,*)
```

```
data: Do ix = 1, 3
        x = real(ix,kind=nag_wp)*0.5E0_nag_wp

        Do iy = ix, 3
          y = real(iy,kind=nag_wp)*0.5E0_nag_wp
          z = 1.0E0_nag_wp

          ifail = -1
          rd = s21bcf(x,y,z,ifail)

          If (ifail<0) Then
            Exit data
          End If

          Write (nout,99999) x, y, z, rd
        End Do

      End Do data

99999 Format (1X,3F7.2,F12.4)
    End Program s21bcfe
```

10.2 Program Data

None.

10.3 Program Results

```
 S21BCF Example Program Results

     X      Y      Z        S21BCF

    0.50   0.50   1.00      1.4787
    0.50   1.00   1.00      1.2108
    0.50   1.50   1.00      1.0611
    1.00   1.00   1.00      1.0000
    1.00   1.50   1.00      0.8805
    1.50   1.50   1.00      0.7775
```