NAG Library Routine Document

S21BHF

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

S21BHF returns a value of the classical (Legendre) form of the complete elliptic integral of the first kind, via the function name.

2 Specification

```
FUNCTION S21BHF (DM, IFAIL)
REAL (KIND=nag_wp) S21BHF

INTEGER            IFAIL
REAL (KIND=nag_wp) DM
```

3 Description

S21BHF calculates an approximation to the integral

$$K(m) = \int_0^{\frac{\pi}{2}} \left(1 - m\sin^2\theta\right)^{-\frac{1}{2}} d\theta,$$

where $m < 1$.

The integral is computed using the symmetrised elliptic integrals of Carlson (Carlson (1979) and Carlson (1988)). The relevant identity is

$$K(m) = R_F(0, 1-m, 1),$$

where R_F is the Carlson symmetrised incomplete elliptic integral of the first kind (see S21BBF).

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 Parameters

1: DM – REAL (KIND=nag_wp) *Input*

On entry: the argument m of the function.

Constraint: DM < 1.0.

2: IFAIL – INTEGER *Input/Output*

On entry: IFAIL must be set to 0, -1 or 1. If you are unfamiliar with this parameter you should refer to Section 3.3 in the Essential Introduction 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 parameter, 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, DM $= \langle value \rangle$; the integral is undefined.
Constraint: DM < 1.0.
On softfailure, the routine returns zero.

IFAIL $= 2$

On entry, DM $= 1.0$; the integral is infinite.
On softfailure, the routine returns the largest machine number (see X02ALF).

IFAIL $= -99$

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

See Section 3.8 in the Essential Introduction for further information.

IFAIL $= -399$

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

See Section 3.7 in the Essential Introduction for further information.

IFAIL $= -999$

Dynamic memory allocation failed.

See Section 3.6 in the Essential Introduction for further information.

7 Accuracy

In principle S21BHF 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

Not applicable.

9 Further Comments

You should consult the S Chapter Introduction, which shows the relationship between this routine and the Carlson definitions of the elliptic integrals. In particular, the relationship between the argument-constraints for both forms becomes clear.

For more information on the algorithm used to compute R_F, see the routine document for S21BBF.

10 Example

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

10.1 Program Text

```
    Program s21bhfe

!     S21BHF Example Program Text

!     Mark 25 Release. NAG Copyright 2014.

!     .. Use Statements ..
      Use nag_library, Only: nag_wp, s21bhf
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter               :: nout = 6
!     .. Local Scalars ..
      Real (Kind=nag_wp)               :: dm, f
      Integer                          :: ifail, ix
!     .. Intrinsic Procedures ..
      Intrinsic                        :: real
!     .. Executable Statements ..
      Write (nout,*) 'S21BHF Example Program Results'

      Write (nout,*)
      Write (nout,*) '    DM       S21BHF'
      Write (nout,*)

data: Do ix = 1, 3
        dm = real(ix,kind=nag_wp)*0.25E0_nag_wp

        ifail = -1
        f = s21bhf(dm,ifail)

        If (ifail<0) Then
          Exit data
        End If

        Write (nout,99999) dm, f
      End Do data

99999 Format (1X,F7.2,F12.4)
    End Program s21bhfe
```

10.2 Program Data

None.

10.3 Program Results

```
 S21BHF Example Program Results

     DM       S21BHF

    0.25      1.6858
    0.50      1.8541
    0.75      2.1565
```