

NAG Library Routine Document

S22BAF

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

S22BAF returns a value for the confluent hypergeometric function ${}_1F_1(a; b; x)$ with real parameters a , b and x . This function is sometimes also known as Kummer's function $M(a, b, x)$.

2 Specification

SUBROUTINE S22BAF (A, B, X, M, IFAIL)

INTEGER IFAIL
REAL (KIND=nag_wp) A, B, X, M

3 Description

S22BAF returns a value for the confluent hypergeometric function ${}_1F_1(a; b; x)$ with real parameters a , b and x . This function is unbounded or not uniquely defined for b equal to zero or a negative integer.

The associated routine S22BBF performs the same operations, but returns M in the scaled form $M = m_f \times 2^{m_s}$ to allow calculations to be performed when M is not representable as a single working precision number. It also accepts the parameters a and b as summations of integer and a decimal fraction, giving higher accuracy when a or b are close to an integer. In such cases, S22BBF should be used when high accuracy is required.

The confluent hypergeometric function is defined by the confluent series

$${}_1F_1(a; b; x) = M(a, b, x) = \sum_{s=0}^{\infty} \frac{(a)_s x^s}{(b)_s s!} = 1 + \frac{a}{b}x + \frac{a(a+1)}{b(b+1)2!}x^2 + \dots$$

where $(a)_s = 1(a)(a+1)(a+2)\dots(a+s-1)$ is the rising factorial of a . $M(a, b, x)$ is a solution to the second order ODE (Kummer's Equation):

$$x \frac{d^2 M}{dx^2} + (b-x) \frac{dM}{dx} - aM = 0. \quad (1)$$

Given the parameters (a, b, x) , this routine determines a set of safe parameters $\{(\alpha_i, \beta_i, \zeta_i) \mid i \leq 2\}$ and selects an appropriate algorithm to accurately evaluate the functions $M_i(\alpha_i, \beta_i, \zeta_i)$. The result is then used to construct the solution to the original problem $M(a, b, x)$ using, where necessary, recurrence relations and/or continuation.

Additionally, an artificial bound, *arwnd* is placed on the magnitudes of a , b and x to minimize the occurrence of overflow in internal calculations. $arwnd = 0.0001 \times I_{\max}$, where $I_{\max} = X02BBF$. It should, however, not be assumed that this routine will produce an accurate result for all values of a , b and x satisfying this criterion.

Please consult the NIST Digital Library of Mathematical Functions or the companion (2010) for a detailed discussion of the confluent hypergeometric function including special cases, transformations, relations and asymptotic approximations.

4 References

NIST Handbook of Mathematical Functions (2010) (eds F W J Olver, D W Lozier, R F Boisvert, C W Clark) Cambridge University Press

Pearson J (2009) Computation of hypergeometric functions *MSc Dissertation, Mathematical Institute, University of Oxford*

5 Parameters

1: A – REAL (KIND=nag_wp) *Input*

On entry: the parameter a of the function.

Constraint: $|A| \leq \text{arwnd}$.

2: B – REAL (KIND=nag_wp) *Input*

On entry: the parameter b of the function.

Constraint: $|B| \leq \text{arwnd}$.

3: X – REAL (KIND=nag_wp) *Input*

On entry: the argument x of the function.

Constraint: $|X| \leq \text{arwnd}$.

4: M – REAL (KIND=nag_wp) *Output*

On exit: the solution $M(a, b, x)$.

Note: if overflow occurs upon completion, as indicated by $\text{IFAIL} = 2$, $|M(a, b, x)|$ may be assumed to be too large to be representable. M will be returned as $\pm R_{\max}$, where R_{\max} is the largest representable real number (see X02ALF). The sign of M should match the sign of $M(a, b, x)$. If overflow occurs during a subcalculation, as indicated by $\text{IFAIL} = 5$, the sign may be incorrect, and the true value of $M(a, b, x)$ may or may not be greater than R_{\max} . In either case it is advisable to subsequently use S22BBF.

5: 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: $\text{IFAIL} = 0$ unless the routine detects an error or a warning has been flagged (see Section 6).

6 Error Indicators and Warnings

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

Errors or warnings detected by the routine:

$\text{IFAIL} = 1$

Underflow occurred during the evaluation of $M(a, b, x)$.
The returned value may be inaccurate.

IFAIL = 2

On completion, overflow occurred in the evaluation of $M(a, b, x)$.

IFAIL = 3

All approximations have completed, and the final residual estimate indicates some precision may have been lost.

Relative residual = $\langle value \rangle$.

IFAIL = 4

All approximations have completed, and the final residual estimate indicates no accuracy can be guaranteed.

Relative residual = $\langle value \rangle$.

IFAIL = 5

Overflow occurred in a subcalculation of $M(a, b, x)$.
The answer may be completely incorrect.

IFAIL = 11

On entry, A = $\langle value \rangle$.

Constraint: $|A| \leq arbnd = \langle value \rangle$.

IFAIL = 31

On entry, B = $\langle value \rangle$.

Constraint: $|B| \leq arbnd = \langle value \rangle$.

IFAIL = 32

On entry, B = $\langle value \rangle$.

$M(a, b, x)$ is undefined when b is zero or a negative integer.

IFAIL = 51

On entry, X = $\langle value \rangle$.

Constraint: $|X| \leq arbnd = \langle value \rangle$.

7 Accuracy

In general, if IFAIL = 0, the value of M may be assumed accurate, with the possible loss of one or two decimal places. Assuming the result does not under or overflow, an error estimate res is made internally using equation (1). If the magnitude of res is sufficiently large, a nonzero IFAIL will be returned. Specifically,

$$IFAIL = \left\{ \begin{array}{ll} 0 & res \leq 1000\epsilon \\ 3 & 1000\epsilon < res \leq 0.1 \\ 4 & res > 0.1 \end{array} \right\}.$$

A further estimate of the residual can be constructed using equation (1), and the differential identity,

$$\frac{dM(a,b,x)}{dx} = \frac{a}{b}M(a+1, b+1, x),$$

$$\frac{d^2M(a,b,x)}{dx^2} = \frac{a(a+1)}{b(b+1)}M(a+2, b+2, x).$$

This estimate is however dependent upon the error involved in approximating $M(a+1, b+1, x)$ and $M(a+2, b+2, x)$.

Furthermore, the accuracy of the solution, and the error estimate, can be dependent upon the accuracy of the decimal fraction of the input parameters a and b . For example, if $b = b_i + b_r = 100 + 1.0E-6$, then

on a machine with 16 decimal digits of precision, the internal calculation of b_r will only be accurate to 8 decimal places. This can subsequently pollute the final solution by several decimal places without affecting the residual estimate as greatly. Should you require higher accuracy in such regions, then you should use S22BBF, which requires you to supply the correct decimal fraction.

8 Further Comments

None.

9 Example

This example reads in values for a , b and x and prints the result of S22BAF at these values.

9.1 Program Text

```

Program s22baf

!      S22BAF Example Program Text

!      Mark 24 Release. NAG Copyright 2012.

!      .. Use Statements ..
Use nag_library, Only: nag_wp, s22baf
!      .. Implicit None Statement ..
Implicit None
!      .. Parameters ..
Integer, Parameter          :: nin = 5, nout = 6
!      .. Local Scalars ..
Real (Kind=nag_wp)         :: a, b, m, x
Integer                    :: ifail, istat_read
!      .. Executable Statements ..
Write (nout,*) 'S22BAF Example Program Results'

Read (nin,*)
Read (nin,*) a, b

Write (nout,99999) 'a      ', 'b      '
Write (nout,99998)
Write (nout,99997) a, b
Write (nout,99998)
Write (nout,99994) 'x      ', 'M(a,b,x) ', 'IFAIL      '
Write (nout,99995)

readx: Do
  Read (nin,*,Iostat=istat_read) x
  If (istat_read/=0) Exit readx
  ifail = -1
  Call s22baf(a,b,x,m,ifail)
  Write (nout,99996) x, m, ifail
End Do readx

99999 Format (/2(1X,A14))
99998 Format (2('+-----'),'+')
99997 Format (2(1X,E13.5,1X))
99996 Format (1X,F10.2,'      ',1X,E13.5,1X,I9)
99995 Format (3('+-----'),'+')
99994 Format (/3(1X,A14))
End Program s22baf

```

9.2 Program Data

S22BAF Example Program Data

```

13.6, 14.2 :Values of A, B

-2.50
-0.90

```

```

-0.25
-0.01
 0.10
 0.50
 0.75      :End of X Values

```

9.3 Program Results

S22BAF Example Program Results

```

          a                b
+-----+-----+
 0.13600E+02    0.14200E+02
+-----+-----+

          x                M(a,b,x)                IFAIL
+-----+-----+-----+
 -2.50          0.92072E-01                0
 -0.90          0.42280E+00                0
 -0.25          0.78714E+00                0
 -0.01          0.99047E+00                0
  0.10          0.11005E+01                0
  0.50          0.16148E+01                0
  0.75          0.20525E+01                0

```
