NAG Library Routine Document

C05BAF

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

C05BAF returns the real values of Lambert's W function $W(x)$, via the routine name.

2 Specification

```
FUNCTION C05BAF (X, BRANCH, OFFSET, IFAIL)
REAL (KIND=nag_wp) C05BAF
INTEGER            BRANCH, IFAIL
REAL (KIND=nag_wp) X
LOGICAL            OFFSET
```

3 Description

C05BAF calculates an approximate value for the real branches of Lambert's W function (sometimes known as the 'product log' or 'Omega' function), which is the inverse function of

$$f(w) = we^w \quad \text{for} \quad w \in C.$$

The function f is many-to-one, and so, except at 0, W is multivalued. C05BAF restricts W and its argument x to be real, resulting in a function defined for $x \geq -\exp(-1)$ and which is double valued on the interval $(-\exp(-1), 0)$. This double-valued function is split into two real-valued branches according to the sign of $W(x) + 1$. We denote by W_0 the branch satisfying $W_0(x) \geq -1$ for all real x, and by W_{-1} the branch satisfying $W_{-1}(x) \leq -1$ for all real x. You may select your branch of interest using the argument BRANCH.

The precise method used to approximate W is described fully in Barry *et al.* (1995). For x close to $-\exp(-1)$ greater accuracy comes from evaluating $W(-\exp(-1) + \Delta x)$ rather than $W(x)$: by setting OFFSET = .TRUE. on entry you inform C05BAF that you are providing Δx, not x, in X.

4 References

Barry D J, Culligan–Hensley P J, and Barry S J (1995) Real values of the W-function *ACM Trans. Math. Software* **21(2)** 161–171

5 Arguments

1: X – REAL (KIND=nag_wp) *Input*

On entry: if OFFSET = .TRUE., X is the offset Δx from $-\exp(-1)$ of the intended argument to W; that is, $W(\beta)$ is computed, where $\beta = -\exp(-1) + \Delta x$.

If OFFSET = .FALSE., X is the argument x of the function; that is, $W(\beta)$ is computed, where $\beta = x$.

Constraints:

if BRANCH = 0, $-\exp(-1) \leq \beta$;
if BRANCH = -1, $-\exp(-1) \leq \beta < 0.0$.

2: BRANCH – INTEGER *Input*

On entry: the real branch required.

BRANCH $= 0$
The branch W_0 is selected.

BRANCH $= -1$
The branch W_{-1} is selected.

Constraint: BRANCH $= 0$ or -1.

3: OFFSET – LOGICAL *Input*

On entry: controls whether or not X is being specified as an offset from $-\exp(-1)$.

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, because for this routine the values of the output arguments may be useful even if IFAIL $\neq 0$ on exit, the recommended value is -1. **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).

Note: C05BAF may return useful information for one or more of the following detected errors or warnings.

Errors or warnings detected by the routine:

IFAIL $= 1$

On entry, BRANCH $= \langle value \rangle$.
Constraint: BRANCH $= 0$ or -1.

On entry, BRANCH $= -1$, OFFSET $=$ .FALSE. and X $= \langle value \rangle$.
Constraint: if BRANCH $= -1$ and OFFSET $=$ .FALSE. then X < 0.0.

On entry, BRANCH $= -1$, OFFSET $=$ .TRUE. and X $= \langle value \rangle$.
Constraint: if BRANCH $= -1$ and OFFSET $=$ .TRUE. then X $< \exp(-1.0)$.

On entry, OFFSET $=$ .FALSE. and X $= \langle value \rangle$.
Constraint: if OFFSET $=$ .FALSE. then X $\geq -\exp(-1.0)$.

On entry, OFFSET $=$ .TRUE. and X $= \langle value \rangle$.
Constraint: if OFFSET $=$ .TRUE. then X ≥ 0.0.

IFAIL $= 2$

For the given offset X, W is negligibly different from -1: X $= \langle value \rangle$.

X is close to $-\exp(-1)$. Enter X as an offset to $-\exp(-1)$ for greater accuracy: X $= \langle value \rangle$.

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

For a high percentage of legal X on input, C05BAF is accurate to the number of decimal digits of precision on the host machine (see X02BEF). An extra digit may be lost on some implementations and for a small proportion of such X. This depends on the accuracy of the base-10 logarithm on your system.

8 Parallelism and Performance

C05BAF is not threaded in any implementation.

9 Further Comments

None.

10 Example

This example reads from a file the values of the required branch, whether or not the arguments to W are to be considered as offsets to $-\exp(-1)$, and the arguments X themselves. It then evaluates the function for these sets of input data X and prints the results.

10.1 Program Text

```
    Program c05bafe

!     C05BAF Example Program Text

!     Mark 26 Release. NAG Copyright 2016.

!     .. Use Statements ..
      Use nag_library, Only: c05baf, nag_wp
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter               :: nin = 5, nout = 6
!     .. Local Scalars ..
      Real (Kind=nag_wp)               :: w, x
      Integer                          :: branch, ifail, ioerr
      Logical                          :: offset
!     .. Executable Statements ..
      Write (nout,*) 'C05BAF Example Program Results'

!     Skip heading in data file
      Read (nin,*)

      Read (nin,*) branch
      Read (nin,*) offset

      Write (nout,*)
      Write (nout,99998) 'BRANCH = ', branch

      If (offset) Then
        Write (nout,99997) 'OFFSET = .TRUE.'
```

```
      Else
        Write (nout,99997) 'OFFSET = .FALSE.'
      End If

      Write (nout,*)
      Write (nout,*) '        X           W(X)     IFAIL'
      Write (nout,*)

data: Do
        Read (nin,*,Iostat=ioerr) x

        If (ioerr<0) Then
          Exit data
        End If

        ifail = -1
        w = c05baf(x,branch,offset,ifail)

        If (ifail<0) Then
          Exit data
        End If

        Write (nout,99999) x, w, ifail
      End Do data

99999 Format (1X,1P,2(1X,E13.5),1X,I5)
99998 Format (1X,A,I3)
99997 Format (1X,A)
    End Program c05bafe
```

10.2 Program Data

```
C05BAF Example Program Data
  0                                                     : BRANCH
  .FALSE.                                               : OFFSET
  0.5
  1.0
  4.5
  6.0
  7.0D7                                                 : X
```

10.3 Program Results

```
 C05BAF Example Program Results

 BRANCH =   0
 OFFSET = .FALSE.

          X           W(X)     IFAIL

    5.00000E-01   3.51734E-01     0
    1.00000E+00   5.67143E-01     0
    4.50000E+00   1.26724E+00     0
    6.00000E+00   1.43240E+00     0
    7.00000E+07   1.53339E+01     0
```