

NAG Library Routine Document

S17DHF

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

S17DHF returns the value of the Airy function $\text{Bi}(z)$ or its derivative $\text{Bi}'(z)$ for complex z , with an option for exponential scaling.

2 Specification

```
SUBROUTINE S17DHF (DERIV, Z, SCAL, BI, IFAIL)
  INTEGER          IFAIL
  COMPLEX (KIND=nag_wp) Z, BI
  CHARACTER(1)    DERIV, SCAL
```

3 Description

S17DHF returns a value for the Airy function $\text{Bi}(z)$ or its derivative $\text{Bi}'(z)$, where z is complex, $-\pi < \arg z \leq \pi$. Optionally, the value is scaled by the factor $e^{|\text{Re}(2z\sqrt{z}/3)|}$.

The routine is derived from the routine CBIRY in Amos (1986). It is based on the relations

$\text{Bi}(z) = \frac{\sqrt{z}}{\sqrt{3}}(I_{-1/3}(w) + I_{1/3}(w))$, and $\text{Bi}'(z) = \frac{z}{\sqrt{3}}(I_{-2/3}(w) + I_{2/3}(w))$, where I_ν is the modified Bessel function and $w = 2z\sqrt{z}/3$.

For very large $|z|$, argument reduction will cause total loss of accuracy, and so no computation is performed. For slightly smaller $|z|$, the computation is performed but results are accurate to less than half of *machine precision*. If $\text{Re}(z)$ is too large, and the unscaled function is required, there is a risk of overflow and so no computation is performed. In all the above cases, a warning is given by the routine.

4 References

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

Amos D E (1986) Algorithm 644: A portable package for Bessel functions of a complex argument and non-negative order *ACM Trans. Math. Software* **12** 265–273

5 Arguments

1: DERIV – CHARACTER(1) *Input*

On entry: specifies whether the function or its derivative is required.

DERIV = 'F'

$\text{Bi}(z)$ is returned.

DERIV = 'D'

$\text{Bi}'(z)$ is returned.

Constraint: DERIV = 'F' or 'D'.

2: Z – COMPLEX (KIND=nag_wp) *Input*

On entry: the argument z of the function.

- 3: SCAL – CHARACTER(1) *Input*
On entry: the scaling option.
 SCAL = 'U'
 The result is returned unscaled.
 SCAL = 'S'
 The result is returned scaled by the factor $e^{|\operatorname{Re}(2z\sqrt{z}/3)|}$.
Constraint: SCAL = 'U' or 'S'.
- 4: BI – COMPLEX (KIND=nag_wp) *Output*
On exit: the required function or derivative value.
- 5: 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, DERIV \neq 'F' or 'D'.
 or SCAL \neq 'U' or 'S'.

IFAIL = 2

No computation has been performed due to the likelihood of overflow, because $\operatorname{real}(Z)$ is too large – how large depends on the overflow threshold of the machine. This error exit can only occur when SCAL = 'U'.

IFAIL = 3

The computation has been performed, but the errors due to argument reduction in elementary functions make it likely that the result returned by S17DHF is accurate to less than half of *machine precision*. This error exit may occur if $\operatorname{abs}(Z)$ is greater than a machine-dependent threshold value (given in the Users' Note for your implementation).

IFAIL = 4

No computation has been performed because the errors due to argument reduction in elementary functions mean that all precision in the result returned by S17DHF would be lost. This error exit may occur if $\operatorname{abs}(Z)$ is greater than a machine-dependent threshold value (given in the Users' Note for your implementation).

IFAIL = 5

No result is returned because the algorithm termination condition has not been met. This may occur because the arguments supplied to S17DHF would have caused overflow or underflow.

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

All constants in S17DHF are given to approximately 18 digits of precision. Calling the number of digits of precision in the floating-point arithmetic being used t , then clearly the maximum number of correct digits in the results obtained is limited by $p = \min(t, 18)$. Because of errors in argument reduction when computing elementary functions inside S17DHF, the actual number of correct digits is limited, in general, by $p - s$, where $s \approx \max(1, |\log_{10} |z||)$ represents the number of digits lost due to the argument reduction. Thus the larger the value of $|z|$, the less the precision in the result.

Empirical tests with modest values of z , checking relations between Airy functions $\text{Ai}(z)$, $\text{Ai}'(z)$, $\text{Bi}(z)$ and $\text{Bi}'(z)$, have shown errors limited to the least significant 3 – 4 digits of precision.

8 Parallelism and Performance

S17DHF is not threaded in any implementation.

9 Further Comments

Note that if the function is required to operate on a real argument only, then it may be much cheaper to call S17AHF or S17AKF.

10 Example

This example prints a caption and then proceeds to read sets of data from the input data stream. The first datum is a value for the argument DERIV, the second is a complex value for the argument, Z, and the third is a character value to set the argument SCAL. The program calls the routine and prints the results. The process is repeated until the end of the input data stream is encountered.

10.1 Program Text

```

Program s17dhfe

!      S17DHF Example Program Text

!      Mark 26 Release. NAG Copyright 2016.

!      .. Use Statements ..
!      Use nag_library, Only: nag_wp, s17dhf
!      .. Implicit None Statement ..
!      Implicit None

```

```

! .. Parameters ..
Integer, Parameter      :: nin = 5, nout = 6
! .. Local Scalars ..
Complex (Kind=nag_wp)  :: bi, z
Integer                 :: ifail, ioerr
Character (1)           :: deriv, scal
! .. Executable Statements ..
Write (nout,*) 'S17DHF Example Program Results'

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

Write (nout,*)
Write (nout,*) 'DERIV          Z          SCAL          BI'
Write (nout,*)

data: Do
  Read (nin,*,Iostat=ioerr) deriv, z, scal

  If (ioerr<0) Then
    Exit data
  End If

  ifail = 0
  Call s17dhf(deriv,z,scal,bi,ifail)

  Write (nout,99999) deriv, z, scal, bi
End Do data

99999 Format (3X,A,' (' ,F8.4,',',F8.4,') ',A,' (' ,F8.4,',',F8.4,')')
End Program s17dhfe

```

10.2 Program Data

```

S17DHF Example Program Data
'F' ( 0.3, 0.4) 'U'
'F' ( 0.2, 0.0) 'U'
'F' ( 1.1, -6.6) 'U'
'F' ( 1.1, -6.6) 'S'
'D' (-1.0, 0.0) 'U'

```

10.3 Program Results

S17DHF Example Program Results

DERIV	Z	SCAL	BI
F	(0.3000, 0.4000)	U	(0.7355, 0.1825)
F	(0.2000, 0.0000)	U	(0.7055, 0.0000)
F	(1.1000, -6.6000)	U	(-47.9039, 43.6634)
F	(1.1000, -6.6000)	S	(-0.1300, 0.1185)
D	(-1.0000, 0.0000)	U	(0.5924, 0.0000)
