

NAG Library Routine Document

S18CSF

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

S18CSF returns an array of values of the scaled modified Bessel function $e^{-|x|}I_0(x)$.

2 Specification

```
SUBROUTINE S18CSF (N, X, F, IFAIL)
  INTEGER                N, IFAIL
  REAL (KIND=nag_wp) X(N), F(N)
```

3 Description

S18CSF evaluates an approximation to $e^{-|x_i|}I_0(x_i)$, where I_0 is a modified Bessel function of the first kind for an array of arguments x_i , for $i = 1, 2, \dots, n$. The scaling factor $e^{-|x|}$ removes most of the variation in $I_0(x)$.

The routine uses the same Chebyshev expansions as S18ASF, which returns an array of the unscaled values of $I_0(x)$.

4 References

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

5 Arguments

- | | | |
|----|---|---------------------|
| 1: | N – INTEGER | <i>Input</i> |
| | <i>On entry:</i> n , the number of points. | |
| | <i>Constraint:</i> $N \geq 0$. | |
| 2: | X(N) – REAL (KIND=nag_wp) array | <i>Input</i> |
| | <i>On entry:</i> the argument x_i of the function, for $i = 1, 2, \dots, N$. | |
| 3: | F(N) – REAL (KIND=nag_wp) array | <i>Output</i> |
| | <i>On exit:</i> $e^{- x_i }I_0(x_i)$, the function values. | |
| 4: | IFAIL – INTEGER | <i>Input/Output</i> |
| | <i>On entry:</i> 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. | |
| | <i>On exit:</i> 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, $N = \langle value \rangle$.
Constraint: $N \geq 0$.

$IFAIL = -99$

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

Relative errors in the argument are attenuated when propagated into the function value. When the accuracy of the argument is essentially limited by the *machine precision*, the accuracy of the function value will be similarly limited by at most a small multiple of the *machine precision*.

8 Parallelism and Performance

S18CSF is not threaded in any implementation.

9 Further Comments

None.

10 Example

This example reads values of X from a file, evaluates the function at each value of x_i and prints the results.

10.1 Program Text

```

Program s18csfe

!      S18CSF Example Program Text

!      Mark 26 Release. NAG Copyright 2016.

!      .. Use Statements ..
      Use nag_library, Only: nag_wp, s18csf
!      .. Implicit None Statement ..
      Implicit None
!      .. Parameters ..
      Integer, Parameter          :: nin = 5, nout = 6
!      .. Local Scalars ..

```

```

      Integer                                :: i, ifail, n
!      .. Local Arrays ..
      Real (Kind=nag_wp), Allocatable :: f(:), x(:)
!      .. Executable Statements ..
      Write (nout,*) 'S18CSF Example Program Results'

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

      Write (nout,*)
      Write (nout,*) '      X      F'
      Write (nout,*)

      Read (nin,*) n

      Allocate (x(n),f(n))

      Read (nin,*) x(1:n)

      ifail = 0
      Call s18csf(n,x,f,ifail)

      Do i = 1, n
         Write (nout,99999) x(i), f(i)
      End Do

99999 Format (1X,1P,2E12.3,I5)
      End Program s18csfe

```

10.2 Program Data

S18CSF Example Program Data

8

0.0 0.5 1.0 3.0 6.0 10.0 1000.0 -1.0

10.3 Program Results

S18CSF Example Program Results

X	F
0.000E+00	1.000E+00
5.000E-01	6.450E-01
1.000E+00	4.658E-01
3.000E+00	2.430E-01
6.000E+00	1.667E-01
1.000E+01	1.278E-01
1.000E+03	1.262E-02
-1.000E+00	4.658E-01
