NAG Fortran Library Routine Document

S17AEF

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

S17AEF returns the value of the Bessel Function $J_0(x)$, via the routine name.

2 Specification

```
real FUNCTION S17AEF(X, IFAIL)
INTEGER            IFAIL
real               X
```

3 Description

This routine evaluates an approximation to the Bessel Function of the first kind $J_0(x)$.

Note: $J_0(-x) = J_0(x)$, so the approximation need only consider $x \geq 0$.

The routine is based on three Chebyshev expansions:

For $0 < x \leq 8$,

$$J_0(x) = {\sum_{r=0}}' a_r T_r(t), \quad \text{with } t = 2\left(\frac{x}{8}\right)^2 - 1.$$

For $x > 8$,

$$J_0(x) = \sqrt{\frac{2}{\pi x}}\left\{P_0(x)\cos\left(x - \frac{\pi}{4}\right) - Q_0(x)\sin\left(x - \frac{\pi}{4}\right)\right\},$$

where $P_0(x) = {\sum_{r=0}}' b_r T_r(t)$,

and $Q_0(x) = \frac{8}{x}{\sum_{r=0}}' c_r T_r(t)$,

with $t = 2\left(\frac{8}{x}\right)^2 - 1$.

For x near zero, $J_0(x) \simeq 1$. This approximation is used when x is sufficiently small for the result to be correct to ***machine precision***.

For very large x, it becomes impossible to provide results with any reasonable accuracy (see Section 7), hence the routine fails. Such arguments contain insufficient information to determine the phase of oscillation of $J_0(x)$; only the amplitude, $\sqrt{\frac{2}{\pi|x|}}$, can be determined and this is returned on soft failure. The range for which this occurs is roughly related to the ***machine precision***; the routine will fail if $|x| \gtrsim 1/$***machine precision*** (see the Users' Note for your implementation).

4 References

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

Clenshaw C W (1962) Mathematical tables *Chebyshev-series for Mathematical Functions* HMSO

5 Parameters

1: X – ***real*** *Input*

On entry: the argument x of the function.

2: IFAIL – INTEGER *Input/Output*

On entry: IFAIL must be set to 0, -1 or 1. Users who are unfamiliar with this parameter should refer to Chapter P01 for details.

On exit: IFAIL $= 0$ unless the routine detects an error (see Section 6).

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, for users 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.**

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 $= 1$

X is too large. On soft failure the routine returns the amplitude of the J_0 oscillation, $\sqrt{\frac{2}{\pi|x|}}$.

7 Accuracy

Let δ be the relative error in the argument and E be the absolute error in the result. (Since $J_0(x)$ oscillates about zero, absolute error and not relative error is significant.)

If δ is somewhat larger than the ***machine precision*** (e.g., if δ is due to data errors etc.), then E and δ are approximately related by:

$$E \simeq |xJ_1(x)|\delta$$

(provided E is also within machine bounds). Figure 1 displays the behaviour of the amplification factor $|xJ_1(x)|$.

However, if δ is of the same order as ***machine precision***, then rounding errors could make E slightly larger than the above relation predicts.

For very large x, the above relation ceases to apply. In this region, $J_0(x) \simeq \sqrt{\frac{2}{\pi|x|}}\cos\left(x - \frac{\pi}{4}\right)$. The amplitude $\sqrt{\frac{2}{\pi|x|}}$ can be calculated with reasonable accuracy for all x, but $\cos\left(x - \frac{\pi}{4}\right)$ cannot. If $x - \frac{\pi}{4}$ is written as $2N\pi + \theta$ where N is an integer and $0 \leq \theta < 2\pi$, then $\cos\left(x - \frac{\pi}{4}\right)$ is determined by θ only. If $x \gtrsim \delta^{-1}$, θ cannot be determined with any accuracy at all. Thus if x is greater than, or of the order of, the inverse of the ***machine precision***, it is impossible to calculate the phase of $J_0(x)$ and the routine must fail.

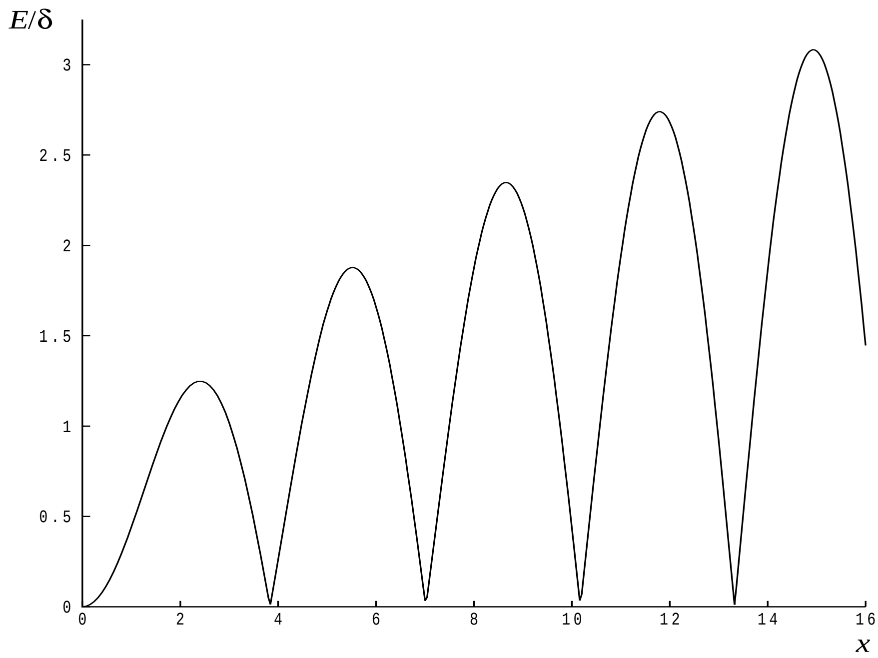

Figure 1

8 Further Comments

None.

9 Example

The example program reads values of the argument x from a file, evaluates the function at each value of x and prints the results.

9.1 Program Text

Note: the listing of the example program presented below uses ***bold italicised*** terms to denote precision-dependent details. Please read the Users' Note for your implementation to check the interpretation of these terms. As explained in the Essential Introduction to this manual, the results produced may not be identical for all implementations.

```
*     S17AEF Example Program Text
*     Mark 14 Revised.  NAG Copyright 1989.
*     .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
*     .. Local Scalars ..
      real             X, Y
      INTEGER          IFAIL
*     .. External Functions ..
      real             S17AEF
      EXTERNAL         S17AEF
*     .. Executable Statements ..
      WRITE (NOUT,*) 'S17AEF Example Program Results'
*     Skip heading in data file
      READ (NIN,*)
      WRITE (NOUT,*)
      WRITE (NOUT,*) '     X           Y        IFAIL'
      WRITE (NOUT,*)
   20 READ (NIN,*,END=40) X
      IFAIL = 1
*
      Y = S17AEF(X,IFAIL)
```

```
*
      WRITE (NOUT,99999) X, Y, IFAIL
      GO TO 20
   40 STOP
*
99999 FORMAT (1X,1P,2e12.3,I7)
      END
```

9.2 Program Data

```
S17AEF Example Program Data
             0.0
             0.5
             1.0
             3.0
             6.0
             8.0
             10.0
             -1.0
             1000.0
```

9.3 Program Results

```
 S17AEF Example Program Results

      X           Y        IFAIL

    0.000E+00   1.000E+00      0
    5.000E-01   9.385E-01      0
    1.000E+00   7.652E-01      0
    3.000E+00  -2.601E-01      0
    6.000E+00   1.506E-01      0
    8.000E+00   1.717E-01      0
    1.000E+01  -2.459E-01      0
   -1.000E+00   7.652E-01      0
    1.000E+03   2.479E-02      0
```