NAG Library Routine Document

S17ADF

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

S17ADF returns the value of the Bessel Function $Y_1(x)$, via the routine name.

2 Specification

```
double precision FUNCTION S17ADF(X, IFAIL)
INTEGER                    IFAIL
double precision           X
```

3 Description

S17ADF evaluates an approximation to the Bessel Function of the second kind $Y_1(x)$.

Note: $Y_1(x)$ is undefined for $x \leq 0$ and the routine will fail for such arguments.

The routine is based on four Chebyshev expansions:

For $0 < x \leq 8$,

$$Y_1(x) = \frac{2}{\pi} \ln x \frac{x}{8} \sum_{r=0}{}' a_r T_r(t) - \frac{2}{\pi x} + \frac{x}{8} \sum_{r=0}{}' b_r T_r(t), \qquad \text{with } t = 2\left(\frac{x}{8}\right)^2 - 1.$$

For $x > 8$,

$$Y_1(x) = \sqrt{\frac{2}{\pi x}} \left\{ P_1(x) \sin\left(x - 3\frac{\pi}{4}\right) + Q_1(x) \cos\left(x - 3\frac{\pi}{4}\right) \right\}$$

where $P_1(x) = \sum_{r=0}{}' c_r T_r(t)$,

and $Q_1(x) = \frac{8}{x} \sum_{r=0}{}' d_r T_r(t)$, with $t = 2\left(\frac{8}{x}\right)^2 - 1$.

For x near zero, $Y_1(x) \simeq -\frac{2}{\pi x}$. This approximation is used when x is sufficiently small for the result to be correct to ***machine precision***. For extremely small x, there is a danger of overflow in calculating $-\frac{2}{\pi x}$ and for such arguments the routine will fail.

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 $Y_1(x)$; 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 ***machine precision***; the routine will fail if $x \gtrsim 1/$***machine precision*** (see the Users' Note for your implementation for details).

4 References

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

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

5 Parameters

1: X – ***double precision*** *Input*

On entry: the argument x of the function.

Constraint: $\mathrm{X} > 0.0$.

2: 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.

On exit: $\mathrm{IFAIL} = 0$ 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, 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.**

6 Error Indicators and Warnings

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

Errors or warnings detected by the routine:

$\mathrm{IFAIL} = 1$

X is too large. On soft failure the routine returns the amplitude of the Y_1 oscillation, $\sqrt{\frac{2}{\pi x}}$.

$\mathrm{IFAIL} = 2$

$\mathrm{X} \leq 0.0$, Y_1 is undefined. On soft failure the routine returns zero.

$\mathrm{IFAIL} = 3$

X is too close to zero, there is a danger of overflow. On soft failure, the routine returns the value of $Y_1(x)$ at the smallest valid argument.

7 Accuracy

Let δ be the relative error in the argument and E be the absolute error in the result. (Since $Y_1(x)$ oscillates about zero, absolute error and not relative error is significant, except for very small x.)

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:

$$E \simeq |xY_0(x) - Y_1(x)|\delta$$

(provided E is also within machine bounds). Figure 1 displays the behaviour of the amplification factor $|xY_0(x) - Y_1(x)|$.

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 small x, absolute error becomes large, but the relative error in the result is of the same order as δ.

For very large x, the above relation ceases to apply. In this region, $Y_1(x) \simeq \frac{2}{\pi x}\sin\left(x - \frac{3\pi}{4}\right)$. The amplitude $\frac{2}{\pi x}$ can be calculated with reasonable accuracy for all x, but $\sin\left(x - \frac{3\pi}{4}\right)$ cannot. If $x - \frac{3\pi}{4}$ is written as $2N\pi + \theta$ where N is an integer and $0 \leq \theta < 2\pi$, then $\sin\left(x - \frac{3\pi}{4}\right)$ is determined by θ only. If

$x > \delta^{-1}$, 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 $Y_1(x)$ and the routine must fail.

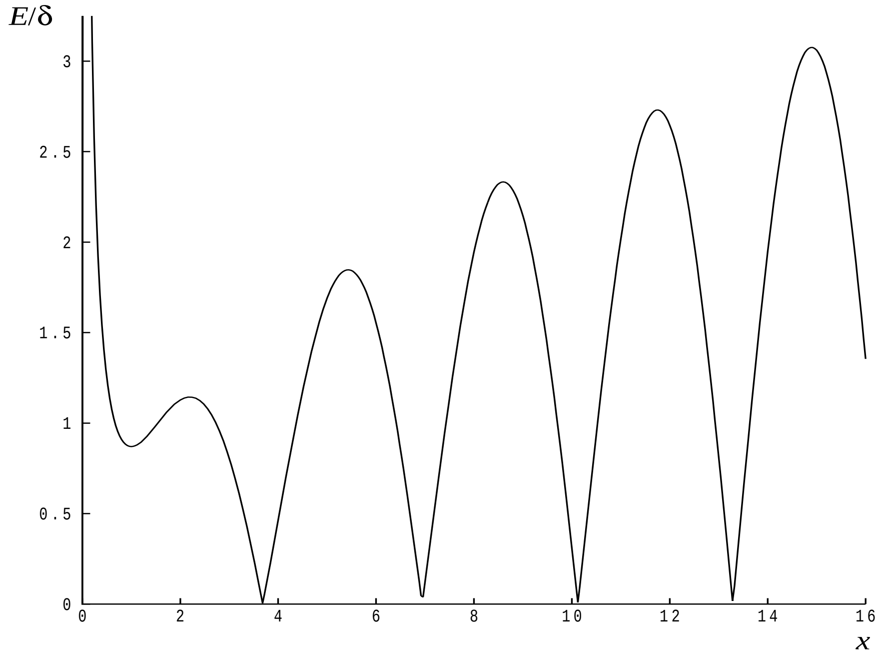

Figure 1

8 Further Comments

None.

9 Example

This example 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

```
*     S17ADF Example Program Text
*     Mark 14 Revised. NAG Copyright 1989.
*     .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
*     .. Local Scalars ..
      DOUBLE PRECISION X, Y
      INTEGER          IFAIL
*     .. External Functions ..
      DOUBLE PRECISION S17ADF
      EXTERNAL         S17ADF
*     .. Executable Statements ..
      WRITE (NOUT,*) 'S17ADF 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 = S17ADF(X,IFAIL)
*
      IF (IFAIL.GE.0) THEN
```

```
         WRITE (NOUT,99999) X, Y, IFAIL
         GO TO 20
      ELSE
         WRITE (NOUT,99998) IFAIL
      END IF
   40 CONTINUE
*
99999 FORMAT (1X,1P,2E12.3,I7)
99998 FORMAT (1X,' ** S17ADF returned with IFAIL = ',I5)
      END
```

9.2 Program Data

```
S17ADF 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

```
 S17ADF Example Program Results

      X           Y        IFAIL

    0.000E+00   0.000E+00      2
    5.000E-01  -1.471E+00      0
    1.000E+00  -7.812E-01      0
    3.000E+00   3.247E-01      0
    6.000E+00  -1.750E-01      0
    8.000E+00  -1.581E-01      0
    1.000E+01   2.490E-01      0
   -1.000E+00   0.000E+00      2
    1.000E+03  -2.478E-02      0
```