NAG Library Routine Document

S17ARF

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

S17ARF returns an array of values of the Bessel function $Y_1(x)$.

2 Specification

```
SUBROUTINE S17ARF (N, X, F, IVALID, IFAIL)

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

3 Description

S17ARF evaluates an approximation to the Bessel function of the second kind $Y_1(x_i)$ for an array of arguments x_i, for $i = 1, 2, \ldots, n$.

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: N – INTEGER *Input*

On entry: n, the number of points.

Constraint: $\mathrm{N} \geq 0$.

2: X(N) – REAL (KIND=nag_wp) array *Input*

On entry: the argument x_i of the function, for $i = 1, 2, \ldots, \mathrm{N}$.

Constraint: $\mathrm{X}(i) > 0.0$, for $i = 1, 2, \ldots, \mathrm{N}$.

3: F(N) – REAL (KIND=nag_wp) array *Output*

On exit: $Y_1(x_i)$, the function values.

4: IVALID(N) – INTEGER array *Output*

On exit: IVALID(i) contains the error code for x_i, for $i = 1, 2, \ldots, \mathrm{N}$.

IVALID(i) = 0
No error.

IVALID(i) = 1

On entry, x_i is too large. F(i) contains the amplitude of the Y_1 oscillation, $\sqrt{\frac{2}{\pi x_i}}$.

IVALID(i) = 2

On entry, $x_i \leq 0.0$, Y_1 is undefined. F(i) contains 0.0.

IVALID(i) = 3
x_i is too close to zero, there is a danger of overflow. On soft failure, F(i) contains the value of $Y_1(x)$ at the smallest valid argument.

5: 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.

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.**

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, at least one value of X was invalid.
Check IVALID for more information.

IFAIL $= 2$

On entry, $\mathrm{N} = \langle value \rangle$.
Constraint: $\mathrm{N} \geq 0$.

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, $Y_1(x) \simeq \sqrt{\frac{2}{\pi x}} \sin\left(x - \frac{3\pi}{4}\right)$. The amplitude $\sqrt{\frac{2}{\pi x}}$ 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.

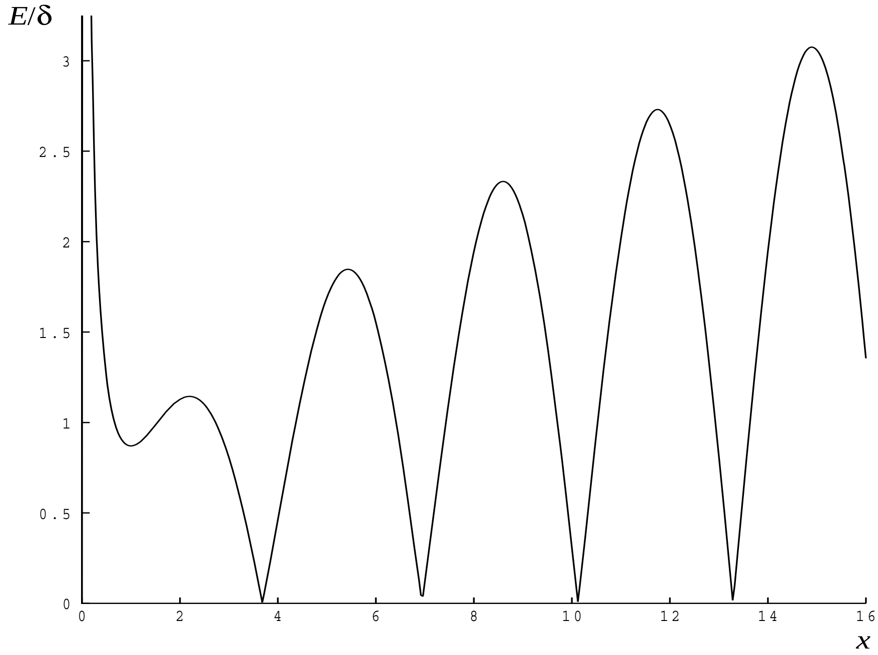

Figure 1

8 Further Comments

None.

9 Example

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

9.1 Program Text

```
    Program s17arfe

!     S17ARF Example Program Text

!     Mark 24 Release. NAG Copyright 2012.

!     .. Use Statements ..
      Use nag_library, Only: nag_wp, s17arf
!     .. 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(:)
      Integer, Allocatable               :: ivalid(:)
!     .. Executable Statements ..
      Write (nout,*) 'S17ARF Example Program Results'

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

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

      Read (nin,*) n

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

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

      ifail = 0
      Call s17arf(n,x,f,ivalid,ifail)

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

99999 Format (1X,1P,2E12.3,I5)
    End Program s17arfe
```

9.2 Program Data

```
S17ARF Example Program Data

7

0.5 1.0 3.0 6.0 8.0 10.0 1000.0
```

9.3 Program Results

```
 S17ARF Example Program Results

      X           F          IVALID

    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+03  -2.478E-02    0
```