

NAG Library Routine Document

S13AAF

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.

Warning. The specification of the parameter X changed at Mark 21: $X < 0.0$ is no longer regarded as an input error.

1 Purpose

S13AAF returns the value of the exponential integral $E_1(x)$, via the routine name.

2 Specification

double precision FUNCTION S13AAF(X, IFAIL)

INTEGER IFAIL

double precision X

3 Description

S13AAF calculates an approximate value for

$$E_1(x) = -\text{Ei}(-x) = \int_x^\infty \frac{e^{-u}}{u} du.$$

using Chebyshev expansions, where x is real. For $x < 0$, the real part of the principal value of the integral is taken. The value $E_1(0)$ is infinite, and so, when $x = 0$, S13AAF exits with an error and returns the largest representable machine number.

For $0 < x \leq 4$,

$$E_1(x) = y(t) - \ln x = \sum_r' a_r T_r(t) - \ln x,$$

where $t = \frac{1}{2}x - 1$.

For $x > 4$,

$$E_1(x) = \frac{e^{-x}}{x} y(t) = \frac{e^{-x}}{x} \sum_r' a_r T_r(t),$$

where $t = -1.0 + \frac{14.5}{(x+3.25)} = \frac{11.25-x}{3.25+x}$.

In both cases, $-1 \leq t \leq +1$.

For $x < 0$, the approximation is based on expansions proposed by Cody and Thatcher Jr. (1969). Precautions are taken to maintain good relative accuracy in the vicinity of $x_0 \approx -0.372507\dots$, which corresponds to a simple zero of $\text{Ei}(-x)$.

S13AAF guards against producing underflows and overflows by using the parameter x_{hi} ; see the Users' Note for your implementation for the value of x_{hi} . To guard against overflow, if $x < -x_{\text{hi}}$ the routine terminates and returns the negative of the largest representable machine number. To guard against underflow, if $x > x_{\text{hi}}$ the result is set directly to zero.

4 References

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

Cody WJ and Thatcher Jr. HC (1969) Rational Chebyshev approximations for the exponential integral $Ei(x)$ *Math. Comp.* **23** 289–303

5 Parameters

1: X – *double precision* *Input*

On entry: the argument x of the function.

Constraint: $-x_{hi} \leq X < 0.0$ or $X > 0.0$.

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: 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 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, $X = 0.0$ and the function is infinite. The result returned is the largest representable machine number.

IFAIL = 2

The evaluation has been abandoned due to the likelihood of overflow. The argument $X < -x_{hi}$, and the result is returned as the negative of the largest representable machine number.

7 Accuracy

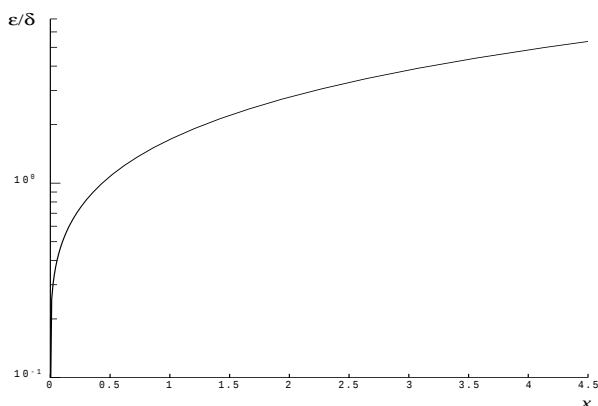
Unless stated otherwise, it is assumed that $x > 0$.

If δ and ϵ are the relative errors in argument and result respectively, then in principle,

$$|\epsilon| \simeq \left| \frac{e^{-x}}{E_1(x)} \times \delta \right|$$

so the relative error in the argument is amplified in the result by at least a factor $e^{-x}/E_1(x)$. The equality should hold if δ is greater than the *machine precision* (δ due to data errors etc.) but if δ is simply a result of round-off in the machine representation, it is possible that an extra figure may be lost in internal calculation and round-off.

The behaviour of this amplification factor is shown in the following graph:



It should be noted that, for absolutely small x , the amplification factor tends to zero and eventually the error in the result will be limited by *machine precision*.

For absolutely large x ,

$$\epsilon \sim x\delta = \Delta,$$

the absolute error in the argument.

For $x < 0$, empirical tests have shown that the maximum relative error is a loss of approximately 1 decimal place.

8 Further Comments

None.

9 Example

The following 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

```
*      S13AAF 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 S13AAF
EXTERNAL          S13AAF
*      .. Executable Statements ..
WRITE (NOUT,*) 'S13AAF 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 = S13AAF(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,' ** S13AAF returned with IFAIL = ',I5)
      END

```

9.2 Program Data

S13AAF Example Program Data

```

      2.0
      0.0
     -1.0
    -1000.0

```

9.3 Program Results

S13AAF Example Program Results

X	Y	IFAIL
2.000E+00	4.890E-02	0
0.000E+00	1.798+308	1
-1.000E+00	-1.895E+00	0
-1.000E+03	-1.798+308	2

