NAG Library Function Document

nag_exp_integral (s13aac)

1 Purpose

nag_exp_integral (s13aac) returns the value of the exponential integral $E_1(x)$.

2 Specification

```
#include <nag.h>
#include <nags.h>
double nag_exp_integral (double x, NagError *fail)
```

3 Description

nag_exp_integral (s13aac) calculates an approximate value for

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

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 $x = 0$, nag_exp_integral (s13aac) 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 $x_0 \approx -0.372507\ldots$, which corresponds to a simple zero of $\mathrm{Ei}(-x)$.

nag_exp_integral (s13aac) guards against producing underflows and overflows by using the argument x_{hi}; see the Users' Note for your implementation for the value of x_{hi}. To guard against overflow, if $x < -x_{\mathrm{hi}}$ the function terminates and returns the negative of the largest representable machine number. To guard against underflow, if $x > x_{\mathrm{hi}}$ 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 W J and Thatcher Jr. H C (1969) Rational Chebyshev approximations for the exponential integral $\mathrm{Ei}(x)$ *Math. Comp.* **23** 289–303

5 Arguments

1: **x** – double *Input*

On entry: the argument x of the function.

Constraint: $-x_{\mathrm{hi}} \leq \mathbf{x} < 0.0$ or $\mathbf{x} > 0.0$.

2: **fail** – NagError * *Input/Output*

The NAG error argument (see Section 3.6 in the Essential Introduction).

6 Error Indicators and Warnings

NE_ALLOC_FAIL

Dynamic memory allocation failed.
See Section 3.2.1.2 in the Essential Introduction for further information.

NE_INTERNAL_ERROR

An internal error has occurred in this function. Check the function call and any array sizes. If the call is correct then please contact NAG for assistance.

An unexpected error has been triggered by this function. Please contact NAG.
See Section 3.6.6 in the Essential Introduction for further information.

NE_NO_LICENCE

Your licence key may have expired or may not have been installed correctly.
See Section 3.6.5 in the Essential Introduction for further information.

NE_REAL_ARG_LE

On entry, $\mathbf{x} = 0.0$ and the function is infinite.

The evaluation has been abandoned due to the likelihood of overflow. The argument $\mathbf{x} < -x_{\mathrm{hi}}$.

7 Accuracy

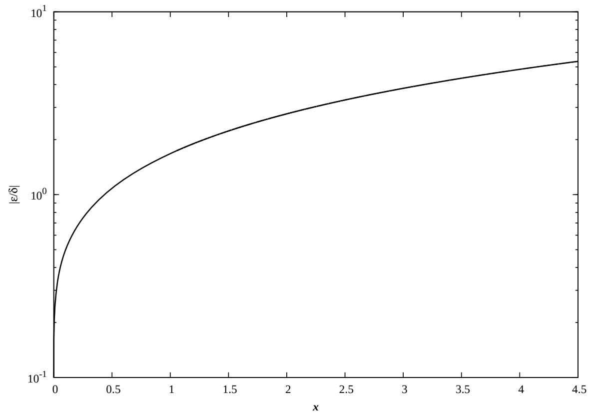
Unless stated otherwise, it is assumed that $x > 0$.

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 Parallelism and Performance

Not applicable.

9 Further Comments

None.

10 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.

10.1 Program Text

```
/* nag_exp_integral (s13aac) Example Program.
 *
 * Copyright 2014 Numerical Algorithms Group.
 *
 * Mark 2 revised, 1992.
 */

#include <nag.h>
#include <stdio.h>
#include <nag_stdlib.h>
#include <nags.h>

int main(void)
{
  Integer  exit_status = 0;
  double   x, y;
  NagError fail;

  INIT_FAIL(fail);
```

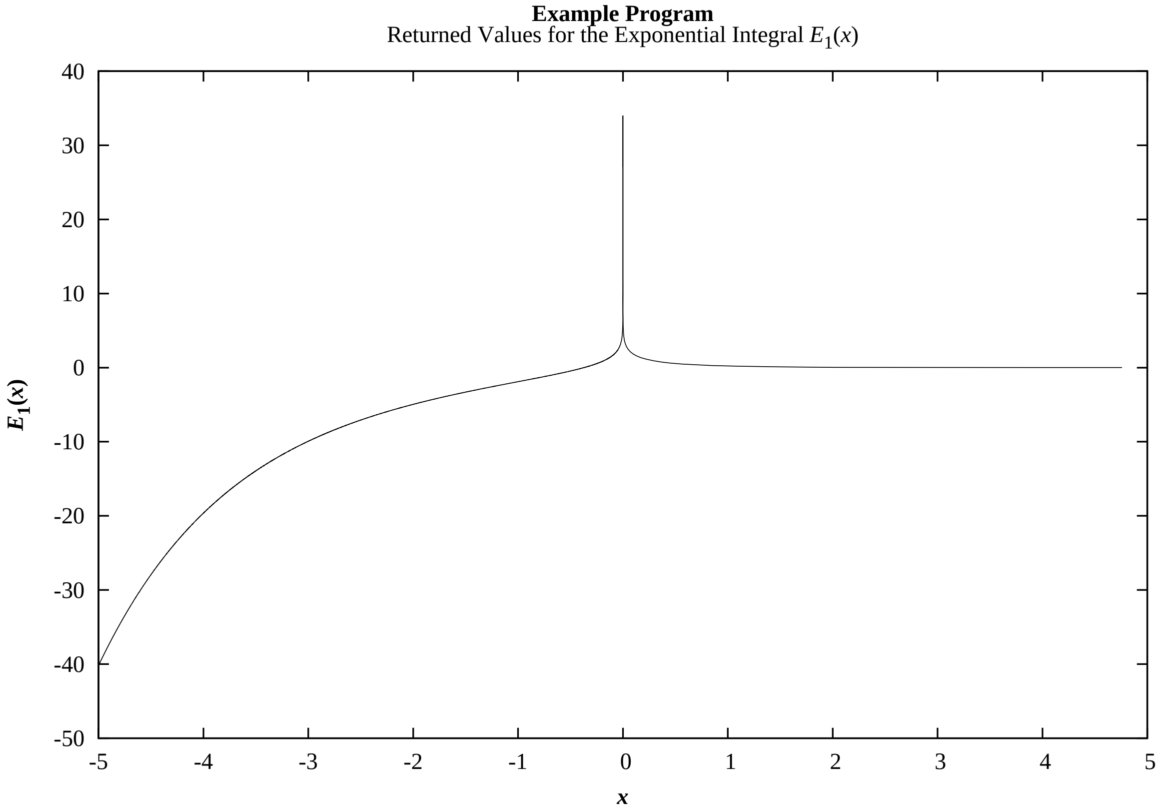
```
  /* Skip heading in data file */
#ifdef _WIN32
  scanf_s("%*[^\n]");
#else
  scanf("%*[^\n]");
#endif
  printf("nag_exp_integral (s13aac) Example Program Results\n");
  printf("     x           y\n");
#ifdef _WIN32
  while (scanf_s("%lf", &x) != EOF)
#else
  while (scanf("%lf", &x) != EOF)
#endif
    {
      /* nag_exp_integral (s13aac).
       * Exponential integral E_1(x)
       */
      y = nag_exp_integral(x, &fail);
      printf("%12.3e%12.3e\n", x, y);
      if (fail.code != NE_NOERROR)
        {
          printf("Error from nag_exp_integral (s13aac).\n%s\n",
                  fail.message);
          if (fail.code != NE_REAL_ARG_LE)
            exit_status = 1;
        }
    }

  return exit_status;
}
```

10.2 Program Data

```
nag_exp_integral (s13aac) Example Program Data
             2.0
             9.0
            -1.0
         -1000.0
```

10.3 Program Results

```
nag_exp_integral (s13aac) Example Program Results
     x           y
   2.000e+00   4.890e-02
   9.000e+00   1.245e-05
  -1.000e+00  -1.895e+00
  -1.000e+03 -1.798e+308
Error from nag_exp_integral (s13aac).
NE_REAL_ARG_LE:
  The evaluation has been abandoned due to the likelihood of overflow.
  The argument x < -x_hi.
```


Example Program
Returned Values for the Exponential Integral $E_1(x)$
40
30
20
10
0
-10
-20
-30
-40
-50
$E_1(x)$
-5
-4
-3
-2
-1
0
1
2
3
4
5
x