1 Purpose

nag_log_gamma (s14abc) returns the value of the logarithm of the gamma function, ln Γ(x).

2 Specification

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

3 Description

nag_log_gamma (s14abc) calculates an approximate value for ln Γ(x). It is based on rational Chebyshev expansions.

Denote by \( R_{n,m}^i(x) = P_n(x)/Q_m(x) \) a ratio of polynomials of degree \( n \) in the numerator and \( m \) in the denominator. Then:

- for \( 0 < x \leq 1/2 \),
  \[
  \ln \Gamma(x) \approx -\ln(x) + xR_{n,m}^1(x + 1);
  \]
- for \( 1/2 < x \leq 3/2 \),
  \[
  \ln \Gamma(x) \approx (x - 1)R_{n,m}^1(x);
  \]
- for \( 3/2 < x \leq 4 \),
  \[
  \ln \Gamma(x) \approx (x - 2)R_{n,m}^2(x);
  \]
- for \( 4 < x \leq 12 \),
  \[
  \ln \Gamma(x) \approx R_{n,m}^3(x);
  \]
- and for \( 12 < x \),
  \[
  \ln \Gamma(x) \approx \left( x - \frac{1}{2} \right) \ln(x) - x + \ln(\sqrt{2\pi}) + \frac{1}{x}R_{n,m}^4(1/x^2).
  \]

For each expansion, the specific values of \( n \) and \( m \) are selected to be minimal such that the maximum relative error in the expansion is of the order \( 10^{-d} \), where \( d \) is the maximum number of decimal digits that can be accurately represented for the particular implementation (see nag_decimal_digits (X02BEC)).

Let \( \epsilon \) denote machine precision and let \( x_{\text{huge}} \) denote the largest positive model number (see nag_real_largest_number (X02ALC)). For \( x < 0.0 \) the value ln Γ(x) is not defined; nag_log_gamma (s14abc) returns zero and exits with fail.code = NE_REAL_ARG_LE. It also exits with fail.code = NE_REAL_ARG_LE when \( x = 0.0 \), and in this case the value \( x_{\text{huge}} \) is returned. For \( x \) in the interval \( (0.0, \epsilon] \), the function ln Γ(x) = −ln(x) to machine accuracy.

Now denote by \( x_{\text{big}} \) the largest allowable argument for ln Γ(x) on the machine. For \( (x_{\text{big}})^{1/4} < x \leq x_{\text{big}} \) the \( R_{n,m}^4(1/x^2) \) term in Equation (1) is negligible. For \( x > x_{\text{big}} \) there is a danger of setting overflow, and so nag_log_gamma (s14abc) exits with fail.code = NE_REAL_ARG_GT and returns \( x_{\text{huge}} \). The value of \( x_{\text{big}} \) is given in the Users’ Note for your implementation.
4 References

5 Arguments
1: \( x \) – double
   \( \text{Input} \)
   \( \text{On entry:} \) the argument \( x \) of the function.
   \( \text{Constraint:} \ x > 0.0. \)
2: \( \text{fail} \) – NagError *
   \( \text{Input/Output} \)
   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_GT**
On entry, \( x = \langle \text{value} \rangle \).
Constraint: \( x \leq x_{\text{big}}. \)

**NE_REAL_ARG_LE**
On entry, \( x = \langle \text{value} \rangle \).
Constraint: \( x > 0.0. \)

7 Accuracy
Let \( \delta \) and \( \epsilon \) be the relative errors in the argument and result respectively, and \( E \) be the absolute error in the result.
If \( \delta \) is somewhat larger than **machine precision**, then
\[
E \approx |x \times \psi(x)|\delta \quad \text{and} \quad \epsilon \approx \left| \frac{x \times \psi(x)}{\ln \Gamma(x)} \right| \delta
\]
where \( \psi(x) \) is the digamma function \( \frac{\Gamma'(x)}{\Gamma(x)} \). Figure 1 and Figure 2 show the behaviour of these error amplification factors.
These show that relative error can be controlled, since except near $x = 1$ or 2 relative error is attenuated by the function or at least is not greatly amplified.

For large $x$, $\epsilon \simeq \left(1 + \frac{1}{\ln x}\right)\delta$ and for small $x$, $\epsilon \simeq \frac{1}{\ln x}\delta$.

The function $\ln \Gamma(x)$ has zeros at $x = 1$ and 2 and hence relative accuracy is not maintainable near those points. However absolute accuracy can still be provided near those zeros as is shown above.

If however, $\delta$ is of the order of machine precision, then rounding errors in the function’s internal arithmetic may result in errors which are slightly larger than those predicted by the equalities. It should be noted that even in areas where strong attenuation of errors is predicted the relative precision is bounded by the effective machine precision.

8 Parallelism and Performance

Not applicable.

9 Further Comments

None.

10 Example

This example 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

```c
/* nag_log_gamma (s14abc) 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);

    printf("nag_log_gamma (s14abc) Example Program Results\n");
    printf(" x y\n");

    while (scanf("%lf", &x) != EOF)
    {
        /* nag_log_gamma (s14abc).
         * Log Gamma function ln(Gamma(x))
         */
        y = nag_log_gamma(x, &fail);
        if (fail.code != NE_NOERROR)
        {
            printf("Error from nag_log_gamma (s14abc).\n",
                    fail.message);
            exit_status = 1;
            goto END;
        }
        printf("%12.3e%12.3e\n", x, y);
    }

END:
    return exit_status;
}
```

## 10.2 Program Data

```
nag_log_gamma (s14abc) Example Program Data
  1.0
  1.25
  1.5
  1.75
  2.0
  5.0
  10.0
  20.0
  1000.0
```
10.3 Program Results

*nag_log_gamma (s14abc)* Example Program Results

<table>
<thead>
<tr>
<th>x</th>
<th>y</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.000e+00</td>
<td>0.000e+00</td>
</tr>
<tr>
<td>1.250e+00</td>
<td>-9.827e-02</td>
</tr>
<tr>
<td>1.500e+00</td>
<td>-1.208e-01</td>
</tr>
<tr>
<td>1.750e+00</td>
<td>-8.440e-02</td>
</tr>
<tr>
<td>2.000e+00</td>
<td>0.000e+00</td>
</tr>
<tr>
<td>5.000e+00</td>
<td>3.178e+00</td>
</tr>
<tr>
<td>1.000e+01</td>
<td>1.280e+01</td>
</tr>
<tr>
<td>2.000e+01</td>
<td>3.934e+01</td>
</tr>
<tr>
<td>1.000e+03</td>
<td>5.905e+03</td>
</tr>
</tbody>
</table>

*Example Program*

Returned Values for the Logarithm of the Gamma Function, $\ln \Gamma(x)$