NAG Library Function Document

nag_elliptic_integral_rd (s21bcc)

1 Purpose

nag_elliptic_integral_rd (s21bcc) returns a value of the symmetrised elliptic integral of the second kind.

2 Specification

```
#include <nag.h>
#include <nags.h>
```

```
double nag_elliptic_integral_rd (double x, double y, double z,
       NagError *fail)
```

3 Description

nag_elliptic_integral_rd (s21bcc) calculates an approximate value for the integral

$$R_D(x,y,z) = \tfrac{3}{2}\int_0^\infty \frac{dt}{\sqrt{(t+x)(t+y)(t+z)^3}}$$

where x, $y \geq 0$, at most one of x and y is zero, and $z > 0$.

The basic algorithm, which is due to Carlson (1979) and Carlson (1988), is to reduce the arguments recursively towards their mean by the rule:

$$\begin{array}{lcl}
x_0 & = & x, y_0 = y, z_0 = z \\
\mu_n & = & (x_n + y_n + 3z_n)/5 \\
X_n & = & (1 - x_n)/\mu_n \\
Y_n & = & (1 - y_n)/\mu_n \\
Z_n & = & (1 - z_n)/\mu_n \\
\lambda_n & = & \sqrt{x_n y_n} + \sqrt{y_n z_n} + \sqrt{z_n x_n} \\
x_{n+1} & = & (x_n + \lambda_n)/4 \\
y_{n+1} & = & (y_n + \lambda_n)/4 \\
z_{n+1} & = & (z_n + \lambda_n)/4
\end{array}$$

For n sufficiently large,

$$\epsilon_n = \max(|X_n|, |Y_n|, |Z_n|) \sim \left(\tfrac{1}{4}\right)^n$$

and the function may be approximated adequately by a fifth order power series

$$R_D(x,y,z) = \quad 3\sum_{m=0}^{n-1} \tfrac{4^{-m}}{(z_m+\lambda_n)\sqrt{z_m}}$$

$$+\frac{4^{-n}}{\sqrt{\mu_n^3}}\left[1 + \frac{3}{7}S_n^{(2)} + \frac{1}{3}S_n^{(3)} + \frac{3}{22}\left(S_n^{(2)}\right)^2 + \frac{3}{11}S_n^{(4)} + \frac{3}{13}S_n^{(2)}S_n^{(3)} + \frac{3}{13}S_n^{(5)}\right]$$

where $S_n^{(m)} = \left(X_n^m + Y_n^m + 3Z_n^m\right)/2m$. The truncation error in this expansion is bounded by $\frac{3\epsilon_n^6}{\sqrt{(1-\epsilon_n)^3}}$ and the recursive process is terminated when this quantity is negligible compared with the ***machine precision***.

The function may fail either because it has been called with arguments outside the domain of definition, or with arguments so extreme that there is an unavoidable danger of setting underflow or overflow.

Note: $R_D(x,x,x) = x^{-3/2}$, so there exists a region of extreme arguments for which the function value is not representable.

4 References

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

Carlson B C (1979) Computing elliptic integrals by duplication *Numerische Mathematik* **33** 1–16

Carlson B C (1988) A table of elliptic integrals of the third kind *Math. Comput.* **51** 267–280

5 Arguments

1: **x** – double *Input*

2: **y** – double *Input*

3: **z** – double *Input*

On entry: the arguments x, y and z of the function.

Constraint: **x**, **y** ≥ 0.0, **z** > 0.0 and only one of **x** and **y** may be zero.

4: **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_EQ

On entry, **x** and **y** are both 0.0.
Constraint: at most one of **x** and **y** is 0.0.
The function is undefined.

NE_REAL_ARG_GE

On entry, $U = \langle value \rangle$, **x** $= \langle value \rangle$, **y** $= \langle value \rangle$ and **z** $= \langle value \rangle$.
Constraint: **x** $< U$ and **y** $< U$ and **z** $< U$.
There is a danger of setting underflow and the function returns zero.

NE_REAL_ARG_LE

On entry, **z** $= \langle value \rangle$.
Constraint: **z** > 0.0.
The function is undefined.

NE_REAL_ARG_LT

On entry, $L = \langle value \rangle$, $\mathbf{x} = \langle value \rangle$, $\mathbf{y} = \langle value \rangle$ and $\mathbf{z} = \langle value \rangle$.
Constraint: $\mathbf{z} \geq L$ and ($\mathbf{x} \geq L$ or $\mathbf{y} \geq L$).
The function is undefined.

On entry, $\mathbf{x} = \langle value \rangle$ and $\mathbf{y} = \langle value \rangle$.
Constraint: $\mathbf{x} \geq 0.0$ and $\mathbf{y} \geq 0.0$.
The function is undefined.

7 Accuracy

In principle the function is capable of producing full ***machine precision***. However round-off errors in internal arithmetic will result in slight loss of accuracy. This loss should never be excessive as the algorithm does not involve any significant amplification of round-off error. It is reasonable to assume that the result is accurate to within a small multiple of the ***machine precision***.

8 Parallelism and Performance

Not applicable.

9 Further Comments

You should consult the s Chapter Introduction which shows the relationship of this function to the classical definitions of the elliptic integrals.

10 Example

This example simply generates a small set of nonextreme arguments which are used with the function to produce the table of low accuracy results.

10.1 Program Text

```
/* nag_elliptic_integral_rd (s21bcc) 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   rd, x, y, z;
  Integer  ix, iy;
  NagError fail;

  INIT_FAIL(fail);

  printf("nag_elliptic_integral_rd (s21bcc) Example Program Results\n");
  printf(
         "      x      y      z       nag_elliptic_integral_rd (s21bcc)  \n");
  for (ix = 1; ix <= 3; ix++)
    {
      x = ix*0.5;
      for (iy = ix; iy <= 3; iy++)
        {
          y = iy*0.5;
          z = 1.0;
```

```
          /* nag_elliptic_integral_rd (s21bcc).
           * Symmetrised elliptic integral of 2nd kind R_D(xyz)
           */
          rd = nag_elliptic_integral_rd(x, y, z, &fail);
          if (fail.code != NE_NOERROR)
            {
              printf(
                      "Error from nag_elliptic_integral_rd (s21bcc).\n%s\n",
                      fail.message);
              exit_status = 1;
              goto END;
            }
          printf(" %7.2f%7.2f%7.2f%12.4f\n", x, y, z, rd);
        }
    }

 END:
  return exit_status;
}
```

10.2 Program Data

None.

10.3 Program Results

```
nag_elliptic_integral_rd (s21bcc) Example Program Results
      x      y      z       nag_elliptic_integral_rd (s21bcc)
    0.50   0.50   1.00      1.4787
    0.50   1.00   1.00      1.2108
    0.50   1.50   1.00      1.0611
    1.00   1.00   1.00      1.0000
    1.00   1.50   1.00      0.8805
    1.50   1.50   1.00      0.7775
```