NAG Toolbox for MATLAB

s21ba

1 Purpose

s21ba returns a value of an elementary integral, which occurs as a degenerate case of an elliptic integral of the first kind, via the function name.

2 Syntax

```
[result, ifail] = s21ba(x, y)
```

3 Description

s21ba calculates an approximate value for the integral

$$R_C(x,y) = \frac{1}{2}\int_0^\infty \frac{dt}{\sqrt{t+x}(t+y)}$$

where $x \geq 0$ and $y \neq 0$.

This function, which is related to the logarithm or inverse hyperbolic functions for $y < x$ and to inverse circular functions if $x < y$, arises as a degenerate form of the elliptic integral of the first kind. If $y < 0$, the result computed is the Cauchy principal value of the integral.

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

$$\begin{aligned} x_0 &= x & y_0 &= y \\ \mu_n &= (x_n + 2y_n)/3, & S_n &= (y_n - x_n)/3\mu_n \\ & & \lambda_n &= y_n + 2\sqrt{x_n y_n} \\ x_{n+1} &= (x_n + \lambda_n)/4, & y_{n+1} &= (y_n + \lambda_n)/4. \end{aligned}$$

The quantity $|S_n|$ for $n = 0, 1, 2, 3, \ldots$ decreases with increasing n, eventually $|S_n| \sim 1/4^n$. For small enough S_n the required function value can be approximated by the first few terms of the Taylor series about the mean. That is

$$R_C(x,y) = \left(1 + \frac{3S_n{}^2}{10} + \frac{S_n{}^3}{7} + \frac{3S_n{}^4}{8} + \frac{9S_n{}^5}{22}\right) \Big/ \sqrt{\mu_n}.$$

The truncation error involved in using this approximation is bounded by $16|S_n|^6/(1 - 2|S_n|)$ and the recursive process is stopped when S_n is small enough for this truncation error to be negligible compared to the ***machine precision***.

Within the domain of definition, the function value is itself representable for all representable values of its arguments. However, for values of the arguments near the extremes the above algorithm must be modified so as to avoid causing underflows or overflows in intermediate steps. In extreme regions arguments are prescaled away from the extremes and compensating scaling of the result is done before returning to the calling program.

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 Parameters

5.1 Compulsory Input Parameters

1: **x – double scalar**
2: **y – double scalar**

The arguments x and y of the function, respectively.

Constraint: $\mathbf{x} \geq 0.0$ and $\mathbf{y} \neq 0.0$.

5.2 Optional Input Parameters

None.

5.3 Input Parameters Omitted from the MATLAB Interface

None.

5.4 Output Parameters

1: **result – double scalar**

The result of the function.

2: **ifail – int32 scalar**

0 unless the function detects an error (see Section 6).

6 Error Indicators and Warnings

Errors or warnings detected by the function:

ifail $= 1$

On entry, $\mathbf{x} < 0.0$; the function is undefined.

ifail $= 2$

On entry, $\mathbf{y} = 0.0$; the function is undefined.

On soft failure the function returns zero.

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 Further Comments

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

9 Example

```
x = 0.5;
y = 1;
[result, ifail] = s21ba(x, y)
```

```
result =
    1.1107
ifail =
           0
```