NAG C Library Function Document

nag_elliptic_integral_f (s21dac)

1 Purpose

nag_elliptic_integral_f (s21dac) returns the value of the general elliptic integral of the second kind $F(z, k', a, b)$ for a complex argument z.

2 Specification

```
Complex nag_elliptic_integral_f (Complex z, double akp, double a, double b,
          NagError *fail)
```

3 Description

This routine evaluates an approximation to the general elliptic integral of the second kind $F(z, k', a, b)$ given by

$$F(z, k', a, b) = \int_0^z \frac{a + b\zeta^2}{(1 + \zeta^2)\sqrt{(1 + \zeta^2)(1 + k'^2\zeta^2)}} d\zeta,$$

where a and b are real parameters, z is a complex argument whose real part is non-negative and k' is a real parameter (the *complementary modulus*). The evaluation of F is based on the Gauss transformation. Further details, in particular for the conformal mapping provided by F, can be found in Bulirsch (1965).

Special values include

$$F(z, k', 1, 1) = \int_0^z \frac{d\zeta}{\sqrt{(1 + \zeta^2)(1 + k'^2\zeta^2)}},$$

or $F_1(z, k')$ (the *elliptic integral of the first kind*) and

$$F(z, k', 1, k'^2) = \int_0^z \frac{\sqrt{1 + k'^2\zeta^2}}{(1 + \zeta^2)\sqrt{1 + \zeta^2}} d\zeta,$$

or $F_2(z, k')$ (the *elliptic integral of the second kind*). Note that the values of $F_1(z, k')$ and $F_2(z, k')$ are equal to $\tan^{-1}(z)$ in the trivial case $k' = 1$.

nag_elliptic_integral_f is derived from a procedure given by Bulirsch (1965).

Constraints are placed on the values of z and k' in order to avoid the possibility of machine overflow.

4 Parameters

1: **z** – Complex *Input*

On entry: the argument z of the function.

Constraints:

$0.0 \leq \mathbf{z}.\text{re} \leq \lambda$,

$|\mathbf{z}.\text{im}| \leq \lambda$, where $\lambda^6 = 1/\text{X02AMC}$.

2: **akp** – double *Input*

On entry: the argument k' of the function.

Constraint: $|\mathbf{akp}| \leq \lambda$.

3: **a** – double *Input*

On entry: the argument a of the function.

4: **b** – double *Input*

On entry: the argument b of the function.

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

The NAG error parameter (see the Essential Introduction).

5 Error Indicators and Warnings

NE_COMPLEX

On entry, **z** = (<*value*>, <*value*>).
Constraints:

$0.0 \leq \mathbf{z}.\mathrm{re} \leq \lambda$,

$|\mathbf{z}.\mathrm{im}| \leq \lambda$, where $\lambda^6 = 1/\mathrm{X02AMC}$.

NE_S21_CONV

The iterative procedure used to evaluate the integral has failed to converge. The result is returned as zero.

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 consult NAG for assistance.

6 Further Comments

6.1 Accuracy

In principle the routine is capable of achieving full relative precision in the computed values. However, the accuracy obtainable in practice depends on the accuracy of the C standard library elementary functions such as atan2 and log.

6.2 References

Bulirsch R (1965) Numerical calculation of elliptic integrals and elliptic functions *Numer. Math.* **7** 76–90

7 See Also

None.

8 Example

The example program evaluates the elliptic integral of the first kind $F_1(z, k')$ given by

$$F_1(z, k') = \int_0^z \frac{d\zeta}{\sqrt{(1+\zeta^2)(1+k'^2\zeta^2)}},$$

where $z = 1.2 + 3.7i$ and $k' = 0.5$, and prints the results.

8.1 Program Text

```
/* nag_elliptic_integral_f (s21dac) Example Program.
 *
 * Copyright 2000 Numerical Algorithms Group.
 *
 * NAG C Library
 *
 * Mark 6, 2000.
 */

#include <stdio.h>

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

int main(void)
{
    const  char  fmt_99999[]  =  "(%4.1f,  %4.1f)  %7.1f  %7.1f  %7.1f    (%12.4e,
%12.4e)\n";
  Complex y, z, z__1;
  double a, akp, b;
  Integer exit_status=0;
  NagError fail;

  INIT_FAIL(fail);
  Vprintf("s21dac Example Program Results\n\n");

  /*  Skip heading in data file */
  Vscanf("%*[^\n] ");
  Vprintf("      z          akp      a       b                 y\n\n");
   while(scanf(" (%lf,%lf) %lf %lf %lf%*[^\n] ", &z.re, &z.im, &akp, &a, &b) !=
EOF )
    {
      z__1 = s21dac (z, akp, a, b, &fail);
      y.re = z__1.re, y.im = z__1.im;
      if (fail.code == NE_NOERROR)
        {
          Vprintf(fmt_99999,z.re,z.im,akp,a,b,y.re,y.im);
        }
      else
        {
          Vprintf("Error from s21dac.\n%s\n", fail.message);
          exit_status = 1;
          goto END;
        }
    }
 END:
  return exit_status;
}
```

8.2 Program Data

```
s21dac Example Program Data
 (1.2, 3.7)  0.5  1.0  1.0
 (9.2,-3.4)  0.8  0.4  2.7  : Values of z, akp, a and b
```

8.3 Program Results

```
s21dac Example Program Results

      z           akp       a       b                  y

( 1.2,  3.7)     0.5     1.0     1.0  (  1.9713e+00,   5.0538e-01)
( 9.2, -3.4)     0.8     0.4     2.7  (  2.5042e+00,  -1.1709e-01)
```