NAG C Library Function Document

nag_jacobian_theta (s21ccc)

1 Purpose

nag_jacobian_theta (s21ccc) returns the value of one of the Jacobian theta functions $\theta_0(x,q)$, $\theta_1(x,q)$, $\theta_2(x,q)$, $\theta_3(x,q)$ or $\theta_4(x,q)$ for a real argument x and non-negative $q \leq 1$.

2 Specification

```
double nag_jacobian_theta (Integer k, double x, double q, NagError *fail)
```

3 Description

This routine evaluates an approximation to the Jacobian theta functions $\theta_0(x,q)$, $\theta_1(x,q)$, $\theta_2(x,q)$, $\theta_3(x,q)$ and $\theta_4(x,q)$ given by

$$\theta_0(x,q) = 1 + 2\sum_{n=1}^{\infty}(-1)^n q^{n^2}\cos(2n\pi x),$$

$$\theta_1(x,q) = 2\sum_{n=0}^{\infty}(-1)^n q^{(n+\frac{1}{2})^2}\sin\{(2n+1)\pi x\},$$

$$\theta_2(x,q) = 2\sum_{n=0}^{\infty} q^{(n+\frac{1}{2})^2}\cos\{(2n+1)\pi x\},$$

$$\theta_3(x,q) = 1 + 2\sum_{n=1}^{\infty} q^{n^2}\cos(2n\pi x),$$

$$\theta_4(x,q) = \theta_0(x,q),$$

where x and q (the *nome*) are real with $0 \leq q \leq 1$. Note that $\theta_1(x-\frac{1}{2},1)$ is undefined if $(x-\frac{1}{2})$ is an integer, as is $\theta_2(x,1)$ if x is an integer; otherwise, $\theta_i(x,1) = 0$, for $i = 0,1,\ldots,4$.

These functions are important in practice because every one of the Jacobian elliptic functions (see nag_jacobian_elliptic (s21cbc)) can be expressed as the ratio of two Jacobian theta functions (see Whittaker and Watson (1990)). There is also a bewildering variety of notations used in the literature to define them. Some authors (e.g., Abramowitz and Stegun (1972), 16.27) define the argument in the trigonometric terms to be x instead of πx. This can often lead to confusion, so great care must therefore be exercised when consulting the literature. Further details (including various relations and identities) can be found in the references.

nag_jacobian_theta (s21ccc) is based on a truncated series approach. If t differs from x or $-x$ by an integer when $0 \leq t \leq \frac{1}{2}$, it follows from the periodicity and symmetry properties of the functions that $\theta_1(x,q) = \pm\theta_1(t,q)$ and $\theta_3(x,q) = \pm\theta_3(t,q)$. In a region for which the approximation is sufficiently accurate, θ_1 is set equal to the first term ($n = 0$) of the transformed series

$$\theta_1(t,q) = 2\sqrt{\frac{\lambda}{\pi}}e^{-\lambda t^2}\sum_{n=0}^{\infty}(-1)^n e^{-\lambda(n+\frac{1}{2})^2}\sinh\{(2n+1)\lambda t\}$$

and θ_3 is set equal to the first two terms (i.e., $n \leq 1$) of

$$\theta_3(t,q) = \sqrt{\frac{\lambda}{\pi}}e^{-\lambda t^2}\left\{1 + 2\sum_{n=1}^{\infty} e^{-\lambda n^2}\cosh(2n\lambda t)\right\},$$

where $\lambda = \pi^2/|\log_e q|$. Otherwise, the trigonometric series for $\theta_1(t,q)$ and $\theta_3(t,q)$ are used. For all values of x, θ_0 and θ_2 are computed from the relations $\theta_0(x,q) = \theta_3(\frac{1}{2} - |x|, q)$ and $\theta_2(x,q) = \theta_1(\frac{1}{2} - |x|, q)$.

4 Parameters

1: **k** – Integer *Input*

On entry: the function $\theta_{\mathrm{K}}(x,q)$ to be evaluated. Note that $\mathbf{k} = 4$ is equivalent to $\mathbf{k} = 0$.

Constraint: $0 \leq \mathbf{k} \leq 4$.

2: **x** – double *Input*

On entry: the argument x of the function.

Constraints:

$\mathbf{x}$ must not be an integer when $\mathbf{q} = 1.0$ and $\mathbf{k} = 2$,

$(\mathbf{x}-0.5)$ must not be an integer when $\mathbf{q} = 1.0$ and $\mathbf{k} = 1$.

3: **q** – double *Input*

On entry: the argument q of the function.

Constraint: $0.0 \leq \mathbf{q} \leq 1.0$.

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

The NAG error parameter (see the Essential Introduction).

5 Error Indicators and Warnings

NE_INT

On entry, $\mathbf{k} =$ <*value*>.
Constraint: $0 \leq \mathbf{k} \leq 4$.

NE_REAL

On entry, $\mathbf{q} =$ <*value*>.
Constraint: $0.0 \leq \mathbf{q} \leq 1.0$.

On entry, $\mathbf{x} =$ <*value*>.
Constraint: $(\mathbf{x}-0.5)$ must not be an integer when $\mathbf{q} = 1.0$ and $\mathbf{k} = 1$.

On entry, $\mathbf{x} =$ <*value*>.
Constraint: $\mathbf{x}$ must not be an integer when $\mathbf{q} = 1.0$ and $\mathbf{k} = 2$.

NE_INFINITE

The evaluation has been abandoned because the function value is infinite.

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 sin and cos.

6.2 References

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

Whittaker E T, and Watson G N (1990) *A Course in Modern Analysis* Cambridge University Press (4th Edition)

Magnus W, Oberhettinger F and Soni R P (1966) *Formulas and Theorems for the Special Functions of Mathematical Physics* 371–377 Springer-Verlag

Tölke F (1966) *Praktische Funktionenlehre (Bd. II)* 1–38 Springer-Verlag

Byrd P F and Friedman M D (1971) *Handbook of Elliptic Integrals for Engineers and Scientists* 315–320 Springer-Verlag (2nd Edition)

7 See Also

None.

8 Example

The example program evaluates $\theta_2(x, q)$ at $x = 0.7$ when $q = 0.4$, and prints the results.

8.1 Program Text

```
/* nag_jacobian_theta (s21ccc) 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[] = "%2ld  %4.1f  %4.1f %12.4e\n";
  double q, x, y;
  Integer exit_status=0;
  Integer k;
  NagError fail;

  INIT_FAIL(fail);
  Vprintf("s21ccc Example Program Results\n\n");

  /* Skip heading in data file */
  Vscanf("%*[^\n] ");
  Vprintf(" k    x     q         y\n\n");
  while (scanf("%ld%lf%lf%*[^\n]", &k, &x, &q) != EOF)
    {
      y = s21ccc  (k, x, q, &fail);
      if (fail.code == NE_NOERROR)
        Vprintf(fmt_99999, k,x,q,y);
      else
        {
```

```
          Vprintf("Error from s21ccc.\n%s\n", fail.message);
          exit_status = 1;
          goto END;
        }
    }
 END:
  return exit_status;
}
```

8.2 Program Data

```
s21ccc Example Program Data
 2  0.7  0.4  : Values of k, x and q
```

8.3 Program Results

```
s21ccc Example Program Results

 k    x     q         y

 2   0.7   0.4  -6.9289e-01
```