

NAG Library Routine Document

S21CAF

Note: before using this routine, please read the Users' Note for your implementation to check the interpretation of ***bold italicised*** terms and other implementation-dependent details.

1 Purpose

S21CAF evaluates the Jacobian elliptic functions sn , cn and dn .

2 Specification

```
SUBROUTINE S21CAF (U, M, SN, CN, DN, IFAIL)
  INTEGER          IFAIL
  REAL (KIND=nag_wp) U, M, SN, CN, DN
```

3 Description

S21CAF evaluates the Jacobian elliptic functions of argument u and argument m ,

$$\begin{aligned}\text{sn}(u \mid m) &= \sin \phi, \\ \text{cn}(u \mid m) &= \cos \phi, \\ \text{dn}(u \mid m) &= \sqrt{1 - m \sin^2 \phi},\end{aligned}$$

where ϕ , called the *amplitude* of u , is defined by the integral

$$u = \int_0^\phi \frac{d\theta}{\sqrt{1 - m \sin^2 \theta}}.$$

The elliptic functions are sometimes written simply as $\text{sn } u$, $\text{cn } u$ and $\text{dn } u$, avoiding explicit reference to the argument m .

Another nine elliptic functions may be computed via the formulae

$$\begin{aligned}\text{cd } u &= \text{cn } u / \text{dn } u \\ \text{sd } u &= \text{sn } u / \text{dn } u \\ \text{nd } u &= 1 / \text{dn } u \\ \text{dc } u &= \text{dn } u / \text{cn } u \\ \text{nc } u &= 1 / \text{cn } u \\ \text{sc } u &= \text{sn } u / \text{cn } u \\ \text{ns } u &= 1 / \text{sn } u \\ \text{ds } u &= \text{dn } u / \text{sn } u \\ \text{cs } u &= \text{cn } u / \text{sn } u\end{aligned}$$

(see Abramowitz and Stegun (1972)).

S21CAF is based on a procedure given by Bulirsch (1960), and uses the process of the arithmetic-geometric mean (16.9 in Abramowitz and Stegun (1972)). Constraints are placed on the values of u and m in order to avoid the possibility of machine overflow.

4 References

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

Bulirsch R (1960) Numerical calculation of elliptic integrals and elliptic functions *Numer. Math.* **7** 76–90

5 Arguments

- 1: U – REAL (KIND=nag_wp) *Input*
 2: M – REAL (KIND=nag_wp) *Input*

On entry: the argument u and the argument m of the functions, respectively.

Constraints:

$$\begin{aligned} \text{abs}(U) &\leq \sqrt{\lambda}, \text{ where } \lambda = 1/\text{X02AMF}; \\ \text{if } \text{abs}(U) < 1/\sqrt{\lambda}, \text{ abs}(M) &\leq \sqrt{\lambda}. \end{aligned}$$

- 3: SN – REAL (KIND=nag_wp) *Output*
 4: CN – REAL (KIND=nag_wp) *Output*
 5: DN – REAL (KIND=nag_wp) *Output*

On exit: the values of the functions $\text{sn } u$, $\text{cn } u$ and $\text{dn } u$, respectively.

- 6: IFAIL – INTEGER *Input/Output*

On entry: IFAIL must be set to 0, -1 or 1. If you are unfamiliar with this argument you should refer to Section 3.4 in How to Use the NAG Library and its Documentation for details.

For environments where it might be inappropriate to halt program execution when an error is detected, the value -1 or 1 is recommended. If the output of error messages is undesirable, then the value 1 is recommended. Otherwise, if you are not familiar with this argument, the recommended value is 0. **When the value -1 or 1 is used it is essential to test the value of IFAIL on exit.**

On exit: IFAIL = 0 unless the routine detects an error or a warning has been flagged (see Section 6).

6 Error Indicators and Warnings

If on entry IFAIL = 0 or -1, explanatory error messages are output on the current error message unit (as defined by X04AAF).

Errors or warnings detected by the routine:

IFAIL = 1

On entry, $\text{abs}(U) > \sqrt{\lambda}$, where $\lambda = 1/\text{X02AMF}()$.

IFAIL = 2

On entry, $\text{abs}(M) > \sqrt{\lambda}$ and $\text{abs}(U) < 1/\sqrt{\lambda}$.

IFAIL = -99

An unexpected error has been triggered by this routine. Please contact NAG.

See Section 3.9 in How to Use the NAG Library and its Documentation for further information.

IFAIL = -399

Your licence key may have expired or may not have been installed correctly.

See Section 3.8 in How to Use the NAG Library and its Documentation for further information.

IFAIL = -999

Dynamic memory allocation failed.

See Section 3.7 in How to Use the NAG Library and its Documentation for further information.

7 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 standard elementary functions such as SIN and COS.

8 Parallelism and Performance

S21CAF is not threaded in any implementation.

9 Further Comments

None.

10 Example

This example reads values of the argument u and argument m from a file, evaluates the function and prints the results.

10.1 Program Text

```

Program s21cafe

!      S21CAF Example Program Text

!      Mark 26 Release. NAG Copyright 2016.

!      .. Use Statements ..
      Use nag_library, Only: nag_wp, s21caf
!      .. Implicit None Statement ..
      Implicit None
!      .. Parameters ..
      Integer, Parameter          :: nin = 5, nout = 6
!      .. Local Scalars ..
      Real (Kind=nag_wp)          :: cn, dn, m, sn, u
      Integer                     :: ifail, ioerr
!      .. Executable Statements ..
      Write (nout,*) 'S21CAF Example Program Results'

!      Skip heading in data file
      Read (nin,*)

      Write (nout,*)
      Write (nout,*)
      '          U              M              SN              CN              DN' &

data: Do
      Read (nin,*,Iostat=ioerr) u, m

      If (ioerr<0) Then
          Exit data
      End If

      ifail = 0
      Call s21caf(u,m,sn,cn,dn,ifail)

      Write (nout,99999) u, m, sn, cn, dn
End Do data

99999 Format (3X,5E13.4)
End Program s21cafe

```

10.2 Program Data

S21CAF Example Program Data

```
0.2   0.3
5.0  -1.0
-0.5  -0.1
10.0  11.0
```

10.3 Program Results

S21CAF Example Program Results

U	M	SN	CN	DN
0.2000E+00	0.3000E+00	0.1983E+00	0.9801E+00	0.9941E+00
0.5000E+01	-0.1000E+01	-0.2440E+00	0.9698E+00	0.1029E+01
-0.5000E+00	-0.1000E+00	-0.4812E+00	0.8766E+00	0.1012E+01
0.1000E+02	0.1100E+02	0.2512E+00	0.9679E+00	0.5528E+00
