NAG Fortran Library Routine Document

S22AAF

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

S22AAF returns a sequence of values for either the unnormalized or normalized Legendre functions of the first kind $P_n^m(x)$ or $\overline{P_n^m}(x)$ for real x of a given order m and degree $n = 0, 1, \ldots, N$.

2 Specification

```
SUBROUTINE S22AAF(MODE, X, M, NL, P, IFAIL)
INTEGER           MODE, M, NL, IFAIL
real              X, P(0:NL)
```

3 Description

This routine evaluates a sequence of values for either the unnormalized or normalized Legendre ($m = 0$) or associated Legendre ($m \neq 0$) functions of the first kind $P_n^m(x)$ or $\overline{P_n^m}(x)$, where x is real with $-1 \leq x \leq 1$, of order m and degree $n = 0, 1, \ldots, N$ defined by

$$\begin{aligned} P_n^m(x) &= (1-x^2)^{m/2}\frac{d^m}{dx^m}P_n(x) && \text{if } m \geq 0, \\ P_n^m(x) &= \frac{(n+m)!}{(n-m)!}P_n^{-m}(x) && \text{if } m < 0 \quad \text{and} \\ \overline{P_n^m}(x) &= \sqrt{\frac{(2n+1)(n-m)!}{2 \quad (n+m)!}}P_n^m(x) \end{aligned}$$

respectively; $P_n(x)$ is the (unassociated) Legendre polynomial of degree n given by

$$P_n(x) \equiv P_n^0(x) = \frac{1}{2^n n!}\frac{d^n}{dx^n}(x^2-1)^n$$

(the *Rodrigues formula*). Note that some authors (e.g., Abramowitz and Stegun (1972)) include an additional factor of $(-1)^m$ (the *Condon-Shortley Phase*) in the definitions of $P_n^m(x)$ and $\overline{P_n^m}(x)$. They use the notation $P_{mn}(x) \equiv (-1)^m P_n^m(x)$ in order to distinguish between the two cases.

S22AAF is based on a standard recurrence relation described in Section 8.5.3 of Abramowitz and Stegun (1972). Constraints are placed on the values of m and n in order to avoid the possibility of machine overflow. It also sets the appropriate elements of the array P (see Section 5) to zero whenever the required function is not defined for certain values of m and n (e.g., $m = -5$ and $n = 3$).

4 References

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

5 Parameters

1: MODE – INTEGER *Input*

On entry: indicates whether the sequence of function values is to be returned unnormalized or normalized as follows:

if MODE $= 1$, then the sequence of function values is returned unnormalized;

if MODE $= 2$, then the sequence of function values is returned normalized.

Constraint: MODE $= 1$ or 2.

2: X – ***real*** *Input*

On entry: the argument x of the function.

Constraint: ABS(X) ≤ 1.0.

3: M – INTEGER *Input*

On entry: the order m of the function.

Constraint: ABS(M) ≤ 27.

4: NL – INTEGER *Input*

On entry: the degree N of the last function required in the sequence.

Constraints:

NL ≥ 0,
NL ≤ 100 when M $= 0$,
NL $\leq 55-$ ABS(M) when M $\neq 0$.

5: P(0:NL) – ***real*** array *Output*

On exit: the required sequence of function values as follows:

if MODE $= 1$, P(n) contains $P_n^m(x)$, for $n = 0, 1, \ldots, N$;

if MODE $= 2$, P(n) contains $\overline{P_n^m}(x)$, for $n = 0, 1, \ldots, N$.

6: IFAIL – INTEGER *Input/Output*

On entry: IFAIL must be set to 0, -1 or 1. Users who are unfamiliar with this parameter should refer to Chapter P01 for details.

On exit: IFAIL $= 0$ unless the routine detects an error (see Section 6).

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, for users not familiar with this parameter the recommended value is 0. **When the value -1 or 1 is used it is essential to test the value of IFAIL on exit.**

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, ABS(X) > 1.0,
or MODE $\neq 1$ or 2,
or NL < 0,
or NL > 100 when M $= 0$,
or ABS(M) > 27,
or NL + ABS(M) > 55 when M $\neq 0$.

7 Accuracy

The computed function values should be accurate to within a small multiple of the ***machine precision*** except when underflow (or overflow) occurs, in which case the true function values are within a small multiple of the underflow (or overflow) threshold of the machine.

8 Further Comments

None.

9 Example

The example program reads the values of the arguments x, m and N from a file, calculates the sequence of unnormalized associated Legendre function values $P_n^m(x), P_{n+1}^m(x), \ldots, P_{n+N}^m(x)$, and prints the results.

9.1 Program Text

Note: the listing of the example program presented below uses ***bold italicised*** terms to denote precision-dependent details. Please read the Users' Note for your implementation to check the interpretation of these terms. As explained in the Essential Introduction to this manual, the results produced may not be identical for all implementations.

```
*     S22AAF Example Program Text.
*     Mark 20 Release. NAG Copyright 2001.
*     .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
      INTEGER          NLMAX
      PARAMETER        (NLMAX=100)
*     .. Local Scalars ..
      real             X
      INTEGER          IFAIL, M, MODE, N, NL
      CHARACTER*80     STR
*     .. Local Arrays ..
      real             P(0:NLMAX)
*     .. External Subroutines ..
      EXTERNAL         S22AAF
*     .. Executable Statements ..
      WRITE (NOUT,*) 'S22AAF Example Program Results'
*     Skip heading in data file
      READ (NIN,*)
      READ (NIN,*) MODE, X, M, NL
      WRITE (NOUT,*)
      IF (MODE.EQ.1) THEN
         IF (M.EQ.0) THEN
            STR = 'Unnormalized Legendre function values'
         ELSE
            STR = 'Unnormalized associated Legendre function values'
         END IF
      ELSE IF (MODE.EQ.2) THEN
         IF (M.EQ.0) THEN
            STR = 'Normalized Legendre function values'
         ELSE
            STR = 'Normalized associated Legendre function values'
         END IF
      END IF
      IFAIL = 0
*
      CALL S22AAF(MODE,X,M,NL,P,IFAIL)
*
      WRITE (NOUT,*) 'MODE      X      M    NL   IFAIL'
      WRITE (NOUT,*)
      WRITE (NOUT,99999) MODE, X, M, NL, IFAIL
      WRITE (NOUT,*)
      WRITE (NOUT,*) STR
      WRITE (NOUT,*) ' n      P(n)'
      DO 20 N = 0, NL
         WRITE (NOUT,99998) N, P(N)
```

```
   20 CONTINUE
*
      STOP
*
99999 FORMAT (1X,I3,4X,F5.1,3I6)
99998 FORMAT (1X,I2,1X,1P,E12.4)
      END
```

9.2 Program Data

```
S22AAF Example Program Data
 1   0.5   2   3  : Values of MODE, X, M and NL
```

9.3 Program Results

```
 S22AAF Example Program Results

 MODE       X       M     NL    IFAIL

   1       0.5      2      3      0

 Unnormalized associated Legendre function values
  n       P(n)
  0   0.0000E+00
  1   0.0000E+00
  2   2.2500E+00
  3   5.6250E+00
```