2.11 Finite Legendre Series

A. Purpose
This subroutine computes the value of a finite sum of Legendre polynomials,

\[ y = \sum_{j=0}^{N} a_j P_j(x) \]

for a specified summation limit, N, argument, x, and sequence of coefficients, \( a_j \). The Legendre polynomials are defined in \[1\].

B. Usage
B.1 Program Prototype, Single Precision

```fortran
INTEGER N
REAL X, Y, A(0 : m \geq N)
Assign values to X, N, and A(0), A(1),... A(N).
CALL SLESUM (S, N, A, Y)
The sum will be stored in Y.
```

B.2 Argument Definitions

- **X** [in] Argument of the polynomials.
- **N** [in] Highest degree of polynomials in sum.
- **A()** [in] The coefficients must be given in A(J), J = 0, ..., N.
- **Y** [out] Computed value of the sum.

B.3 Modifications for Double Precision
For double precision usage, change the REAL statement to DOUBLE PRECISION and change the subroutine name from SLESUM to DLESUM.

C. Examples and Remarks
See DRSLESUM and ODSLESUM for an example of the usage of SLESUM. DRSLESUM evaluates the following identity, the coefficients of which were obtained from Table 22.9, page 798, of \[1\].

\[ z = y - w = 0, \]

where

\[
y = 0.07 P_6(x) + 0.27 P_1(x) + 0.20 P_2(x) \\
+ 0.28 P_3(x) + 0.08 P_4(x) + 0.08 P_5(x),
\]

and

\[
w = 0.35 x^4 + 0.63 x^5.
\]

D. Functional Description
The sum is evaluated by the following algorithm:

\[
b_{N+2} = 0, \quad b_{N+1} = 0, \\
b_k = \frac{2k+1}{k+1} b_{k+1} x - \frac{k+1}{k+2} b_{k+2} + a_k, \quad k = N, ..., 0, \\
y = b_0.
\]

For an error analysis applying to this algorithm see \[2\] and \[3\]. The first four Legendre polynomials are

\[
P_0(x) = 1, \quad P_1(x) = x, \\
P_2(x) = 1.5 x^2 - 0.5, \quad P_3(x) = 2.5 x^3 - 1.5 x.
\]

For \( k \geq 2 \) the Legendre polynomials satisfy the recurrence

\[
k P_k(x) = (2k-1)x P_{k-1}(x) - (k-1) P_{k-2}(x).
\]

The Legendre polynomials are orthogonal relative to integration over the interval \([-1, 1]\) and are normally used only with an argument, \( x \), in this interval.

References

E. Error Procedures and Restrictions
The subroutine will return \( Y = 0 \) if \( N < 0 \). It is recommended that \( x \) satisfy \(| x | \leq 1\).

F. Supporting Information
The source language is ANSI Fortran

**Entry** | **Required Files**
--- | ---
DLESUM | DLESUM
SLESUM | SLESUM

Based on a 1974 program by E. W. Ng, JPL. Present version by C. L. Lawson and S. Y. Chiu, JPL, 1983.
Demonstration driver for evaluation of a Legendre series.

```fortran
integer j
real x, a(0:5), y, w, z
! a = (0.07 e0, 0.27 e0, 0.20 e0, 0.28 e0, 0.08 e0, 0.08 e0) /

print '(1x,3x,a1,14x,a1,17x,a1/) ','x', 'y', 'z'
do 20 j = -10,10,2
   x = real(j) /10.e0
   call slesum (x,5,a,y)
   w = 0.35 e0 * (x**4) + 0.63 e0 * (x**5)
   z = y - w
   print '(1x,f5.2,5x,g15.7,g15.2) ',x,y,z
20 continue
end
```

<table>
<thead>
<tr>
<th>x</th>
<th>y</th>
<th>z</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1.00</td>
<td>-0.2800000</td>
<td>0.0</td>
</tr>
<tr>
<td>-0.80</td>
<td>-0.6307840E-01</td>
<td>0.22E-07</td>
</tr>
<tr>
<td>-0.60</td>
<td>-0.3628805E-02</td>
<td>0.37E-08</td>
</tr>
<tr>
<td>-0.40</td>
<td>0.2508797E-02</td>
<td>-0.33E-08</td>
</tr>
<tr>
<td>-0.20</td>
<td>0.3583953E-03</td>
<td>-0.48E-08</td>
</tr>
<tr>
<td>0.00</td>
<td>0.0000000</td>
<td>0.0</td>
</tr>
<tr>
<td>0.20</td>
<td>0.7616058E-03</td>
<td>0.57E-08</td>
</tr>
<tr>
<td>0.40</td>
<td>0.1541121E-01</td>
<td>0.11E-07</td>
</tr>
<tr>
<td>0.60</td>
<td>0.9434883E-01</td>
<td>0.22E-07</td>
</tr>
<tr>
<td>0.80</td>
<td>0.3497985</td>
<td>0.30E-07</td>
</tr>
<tr>
<td>1.00</td>
<td>0.9800001</td>
<td>0.12E-06</td>
</tr>
</tbody>
</table>