Special Mathematical Functions in Fortran

ISO/IEC 1539-4 : 200x

Auxiliary to ISO/IEC 1539 : 2004 "Programming Language Fortran"

NOTE

This paper is intended to suggest some special functions for which standard procedure interfaces might be specified. Whether it is done as part of Clause 13 of 1539-1, as 1539-4, or as a Technical Report can be decided later. The exact set of procedures can be decided later. Whether the procedures are module procedures or intrinsic procedures can be decided later. If they are module procedures, the module name and whether the module is intrinsic can be decided later.

Subclause 2.4 describes the same procedures as WG14 n1243, plus procedures to compute two additional functions related to the ones described therein that are better behaved.

Subclause 2.5 proposes additional procedures that are widely used in scientific and engineering calculations.

NOTE

At the 2011 WG5 meeting in Garching, there was a discussion (see the Tuesday part of the minutes in N1861) concerning whether WG5 should produce a standard for interfaces to certain mathematical functions, analogous to those for C (ISO/IEC 24747:2009) and C++ (ISO/IEC 29124:2010). The resulting straw vote was: 0 yes - 5 no - 8 undecided.

Contents

Information technology — Programming languages — Fortran —

Part 4:

Special Mathematical Functions

1 Overview

1.1 Scope

ISO/IEC 1539 is a multipart International Standard; the parts are published separately. This publication, ISO/IEC 1539-4, which is the fourth part, describes the standard intrinsic module ISO_Fortran_Special_Functions. The purpose of this part of ISO/IEC 1539 is to promote portability, reliability, maintainability, and efficient evaluation of mathematical special functions in Fortran programs, for use on a variety of computing systems.

This part is normative, but optional. A processor need not provide support for this part.

1.2 Inclusions

This part of ISO/IEC 1539 specifies

- the procedures defined by the module ISO_Fortran_Special_Functions,
- the interface definitions for those procedures, and
- the mathematical function evaluated by each procedure.

1.3 Exclusions

This part of ISO/IEC 1539 does not specify

- the methods to evaluate the functions, or
- the accuracy of the results of the procedures.

1.4 Conformance

A program conforms to ISO/IEC 1539 if it conforms to ISO/IEC 1539-1 and this part of ISO/IEC 1539.

A processor that supports this part of ISO/IEC 1539 conforms to it if

- it executes any standard-conforming program in a manner that fulfills the interpretations herein and in ISO/IEC 1539-1, subject to any limitations that the processor may impose upon the range of the arguments of the procedures, and
- it contains the capability to detect and report the use within a program of argument values outside the ranges specified herein.

1.5 Notation used in this part of ISO/IEC 1539

1.5.1 Applicability of requirements

In this part of ISO/IEC 1539, “shall” is to be interpreted as a requirement; conversely, “shall not” is to be interpreted as a prohibition. Except where stated otherwise, such requirements and prohibitions apply to programs rather than processors.

1.5.2 Informative notes

Informative notes of explanation, rationale, examples, and other material are interspersed with the normative body of this part of ISO/IEC 1539. The informative material is nonnormative; it is identified by being in shaded, framed boxes that have numbered headings beginning with “NOTE.”

1.6 Normative references

The following referenced standards are indispensable for the application of this part of ISO/IEC 1539. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced standard (including any amendments) applies.

ISO/IEC 1539-1:2010, *Information technology—Programming languages—Fortran—Part 1: Base Language*.

ISO 31-11:1992, *Quantities and units—Part 11: Mathematical signs and symbols for use in the physical sciences and technology*, Clause 14 Special Functions.

1.7 Nonnormative references

The following referenced materials are useful but not indispensable for the application of this part of ISO/IEC 1539.

Frank W. J. Olver, Daniel W. Lozier, Ronald F. Boisvert, Charles W. Clark, **NIST Handbook of Mathematical Functions**, National Institute of Standards and Technology and Cambridge University Press (2010), ISBN 978-0-521-19225-5 (hardback), 978-0-521-14063-8 (paperback).

Milton Abramowitz and Irene A. Stegun, **Handbook of Mathematical Functions**, U. S. National Bureau of Standards (now National Institute of Standards and Technology) Applied Mathematics Series #55 (1972) LCCCN 64-60036.

Jerome Spanier and Keith B. Oldham, **An Atlas of Functions**, Hemisphere Publishing Corporation, New York (1987) ISBN 0-89116-573-8.

2 The module ISO_Fortran_Special_Functions

2.1 General

The module ISO_Fortran_Special_Functions contains named mathematical constants and the definitions of the interfaces of procedures to evaluate special mathematical functions. The procedures are all generic procedures. For each generic procedure defined here, the processor shall provide specific procedures for all real kinds supported by the processor. It is processor dependent whether the processor provides specific procedures for integer kinds other than default integer. The names of the specific procedures are private identifiers of ISO_Fortran_Special_Functions. The procedures might be separate module procedures (12.6.2.5 in ISO/IEC 1539-1). If so, the submodule identifiers of the submodules in which the procedures are defined are processor dependent.

It is recommended that documentation that accompanies the processor include descriptions of the relationship between the ranges of the values of the arguments of the procedures and the accuracy of the results.

2.2 Mathematical constants

2.2.1 Euler's constant γ

Euler's constant γ (sometimes called the Euler-Mascheroni constant) is defined as

$$\gamma = \lim_{n\to\infty}\left(\sum_{i=1}^{n}\frac{1}{i} - \ln n\right)$$

and by other definitions that appear in the references (1.6, 1.7).

The kind of the named constant `EULER_GAMMA` shall be the kind supported by the processor that provides the most precise representation. The radix of that kind is processor dependent.

```
REAL(kind), PARAMETER :: EULER_GAMMA = &
    & 0.57721566490153286060651209008240243104215933593992359880576723488486771_kind
```

2.3 Summary of the procedures

ASSOC_LAGUERRE (N, M, X)	Associated Laguerre polynomials
ASSOC_LEGENDRE (L, M, X)	Associated Legendre Polynomials
BETA (X, Y)	Beta function
CYL_BESSEL_I (NU, X)	Regular modified cylindrical Bessel function.
CYL_BESSEL_I (NU, X, N)	Regular modified cylindrical Bessel function.
CYL_BESSEL_J (NU, X)	Cylindrical Bessel function.
CYL_BESSEL_J (NU, X, N)	Cylindrical Bessel function.
CYL_BESSEL_K (NU, X)	Irregular modified cylindrical Bessel function.
CYL_BESSEL_K (NU, X, N)	Irregular modified cylindrical Bessel function.
CYL_NEUMANN (NU, X)	Cylindrical Neumann function.
CYL_NEUMANN (NU, X, N)	Cylindrical Neumann function.

ELLINT_1 (K)	Complete elliptic integral of the first kind
ELLINT_1 (K, PHI)	Incomplete elliptic integral of the first kind
ELLINT_2 (K)	Complete elliptic integral of the second kind
ELLINT_2 (K, PHI)	Incomplete elliptic integral of the second kind
ELLINT_3 (K, NU)	Complete elliptic integral of the third kind
ELLINT_3 (K, NU, PHI)	Complete elliptic integral of the third kind
EIN (X)	Entire exponential integral
EXPINT (X)	Exponential integral
HERMITE (N, X)	Hermite polynomials
LAGUERRE (N, X)	Laguerre polynomials
LEGENDRE (N, X)	Legendre polynomials
RIEMANN_ZETA (X)	Riemann zeta function
SPH_BESSEL (N, X)	Spherical Bessel function of the first kind
SPH_BESSEL (N, X, N)	Spherical Bessel function of the first kind
SPH_LEGENDRE (L, M, THETA)	Spherical associated Legendre function
SPH_NEUMANN (N, X)	Spherical Neumann function
SPH_NEUMANN (N, X, N)	Spherical Neumann function

2.4 Specifications for the procedures

2.4.1 General

Detailed specifications of the procedures whose interfaces are defined in the module ISO_Fortran_Special_-Functions are provided here in alphabetical order.

The types and type parameters of the arguments and function results of these procedures are determined by these specifications. The "Argument(s)" paragraphs specify requirements on the actual arguments of the procedures. The result characteristics are sometimes specified in terms of the characteristics of dummy arguments. A program shall not invoke one of these procedures under circumstances where a value to be assigned to a subroutine argument or returned as a function result is not representable by objects of the specified type and type parameters.

If an IEEE infinity is assigned or returned, the intrinsic module IEEE_ARITHMETIC is accessible, and the actual arguments were finite numbers, the flag IEEE_OVERFLOW or IEEE_DIVIDE_BY_ZERO shall signal. If an IEEE NaN is assigned or returned, the actual arguments were finite numbers, the intrinsic module IEEE_ARITHMETIC is accessible, and the exception IEEE_INVALID is supported, the flag IEEE_INVALID shall signal. If no IEEE infinity or NaN is assigned or returned, these flags shall have the same status as when the intrinsic procedure was invoked.

2.4.2 ASSOC_LAGUERRE (N, M, X)

Description. Associated Laguerre polynomials.

Class. Elemental function.

Arguments.

N	shall be of type integer and nonnegative.
M	shall be of type integer and nonnegative.
X	shall be of type real.

Result Characteristics. Same as X.

Result Value. The value of the result is a processor-dependent approximation to the associated Laguerre polynomial $L_n^m(x)$ of orders N and M and argument X, defined by

$$L_n^m(x) = \frac{1}{n!}\sum_{i=0}^{n}\frac{n!}{i!}\left(\begin{array}{c} m+n \\ n-i \end{array}\right)(-x)^i = (-1)^m \frac{\mathrm{d}^m}{\mathrm{d}x^m} L_{m+n}(x)$$

where $L_{m+n}(x)$ is a Laguerre polynomial (2.4.15)

Example. ASSOC_LAGUERRE (1, 1, 1.0) has the value 1.0 (approximately).

2.4.3 ASSOC_LEGENDRE (L, M, X)

Description. Associated Legendre polynomials.

Class. Elemental function.

Arguments.

L shall be of type integer and nonnegative.

M shall be of type integer and nonnegative.

X shall be of type real. The absolute value of X shall be less than or equal to 1.0.

Result Characteristics. Same as X.

Result Value. The value of the result is a processor-dependent approximation to the associated Legendre polynomial $P_\ell^m(x)$ of orders L and M and argument X, defined by

$$P_\ell^m(x) = (1-x^2)^{m/2} \frac{\mathrm{d}^m}{\mathrm{d}x^m} P_\ell(x), \quad |x| \le 1$$

where $P_\ell(x)$ is a Legendre polynomial (2.4.16).

Example. ASSOC_LEGENDRE (1, 1, 1.0) has the value 0.0 (approximately).

2.4.4 BETA (X, Y)

Description. Beta function.

Class. Elemental function.

Arguments.

X shall be of type real and nonnegative.

Y shall be of type real with the same kind as X. The value of Y shall not be negative.

Result Characteristics. Same as X.

Result Value. The value of the result is a processor-dependent approximation to the beta function $B(x, y)$ with arguments X and Y, defined by

$$B(x, y) = \frac{\Gamma(x)\Gamma(y)}{\Gamma(x+y)} = \int_0^1 t^{x-1}(1-t)^{y-1}\mathrm{d}t, \quad x > 0,\ y > 0$$

and several other representations that appear in the references (1.6, 1.7).

Example. BETA (0.5, 0.5) has the value 3.141592654 (approximately).

2.4.5 CYL_BESSEL_I (NU, X) or CYL_BESSEL_I (NU, X, N)

Description. Regular modified cylindrical Bessel function.

Class.

Case (i): CYL_BESSEL_I (NU, X) is an elemental function.

Case (ii): CYL_BESSEL_I (NU, X, N) is a transformational function.

Arguments.

NU shall be of type real.

X shall be of type real and the same kind as NU. The value of X shall not be negative.

N shall be of type integer and nonnegative.

Result Characteristics. Same type and kind as X.

Case (i): The result of CYL_BESSEL_I (NU, X) is scalar.

Case (ii): The result of CYL_BESSEL_I (NU, X, N) is a rank-one array with extent N.

Result Value.

Case (i): The value of the result of CYL_BESSEL_I (NU, X) is a processor-dependent approximation to the regular modified cylindrical Bessel function $I_\nu(x)$ of order NU with argument X, defined by

$$I_\nu(x) = i^{-\nu} J_\nu(ix) = \left(\frac{x}{2}\right)^\nu \sum_{k=0}^{\infty} \frac{(x/2)^{2k}}{k!\Gamma(\nu+k+1)}, \quad x \geq 0,$$

and several other representations that appear in the references (1.6, 1.7).

Case (ii): The value of the i'th element of the result of CYL_BESSEL_I (NU, X, N) is the same as the value of the result of CYL_BESSEL_I (NU+$i-1$, X)

Example. CYL_BESSEL_I (0.0, 0.0) has the value 1.0 (approximately).

2.4.6 CYL_BESSEL_J (NU, X) or CYL_BESSEL_J (NU, X, N)

Description. Cylindrical Bessel function.

Class.

Case (i): CYL_BESSEL_J (NU, X) is an elemental function.

Case (ii): CYL_BESSEL_J (NU, X, N) is a transformational function.

Arguments.

NU shall be of type real.

X shall be of type real and the same kind as NU. The value of X shall not be negative.

N shall be of type integer and nonnegative.

Result Characteristics. Same type and kind as X.

Case (i): The result of CYL_BESSEL_J (NU, X) is scalar.

Case (ii): The result of CYL_BESSEL_J (NU, X, N) is a rank-one array with extent N.

Result Value.

Case (i): The value of the result of CYL_BESSEL_J (NU, X) is a processor-dependent approximation to the cylindrical Bessel function $J_\nu(x)$ of order NU with argument X, defined by

$$J_\nu(x) = \left(\frac{x}{2}\right)^\nu \sum_{k=0}^{\infty} \frac{(-1)^k (x/2)^{2k}}{k!\Gamma(\nu+k+1)}, \quad x \geq 0,$$

and several other representations that appear in the references (1.6, 1.7).

Case (ii): The value of the i'th element of the result of CYL_BESSEL_J (NU, X, N) is the same as the value of the result of CYL_BESSEL_J (NU+$i-1$, X)

NOTE 2.1

This is a generalization of the standard intrinsic function BESSEL_JN to noninteger order.

Example. CYL_BESSEL_J (0.0, 0.0) has the value 1.0 (approximately).

2.4.7 CYL_BESSEL_K (NU, X) or CYL_BESSEL_K (NU, X, N)

Description. Irregular modified cylindrical Bessel function.

Class.

Case (i): CYL_BESSEL_K (NU, X) is an elemental function.

Case (ii): CYL_BESSEL_K (NU, X, N) is a transformational function.

Arguments.

NU shall be of type real.

X shall be of type real and the same kind as NU. The value of X shall not be negative.

N shall be of type integer and nonnegative.

Result Characteristics. Same type and kind as X.

Case (i): The result of CYL_BESSEL_K (NU, X) is scalar.

Case (ii): The result of CYL_BESSEL_K (NU, X, N) is a rank-one array with extent N.

Result Value.

Case (i): The value of the result of CYL_BESSEL_K (NU, X) is a processor-dependent approximation to the irregular modified cylindrical Bessel function $K_\nu(x)$ of order NU with argument X, defined by

$$K_\nu(x) = \frac{\pi}{2} i^{\nu+1}(J_\nu(ix) + iN_\nu(ix)) = \frac{\pi}{2} \lim_{\mu\to\nu} \frac{I_{-\mu}(x) - I_\mu(x)}{\sin \mu x}, \quad x \geq 0$$

and several other representations that appear in the references (1.6, 1.7).

Case (ii): The value of the i'th element of the result of CYL_BESSEL_K (NU, X, N) is the same as the value of the result of CYL_BESSEL_K (NU+$i-1$, X)

NOTE 2.2

The irregular modified cylindrical Bessel function is also known as the Bassett function.

Example. CYL_BESSEL_K (0.0, HUGE(0.0)) has the value 0.0 (approximately).

2.4.8 CYL_NEUMANN (NU, X) or CYL_NEUMANN (NU, X, N)

Description. Cylindrical Neumann function.

Class.

Case (i): CYL_NEUMANN (NU, X) is an elemental function.

Case (ii): CYL_NEUMANN (NU, X, N) is a transformational function.

Arguments.

NU	shall be of type real.
X	shall be of type real and the same kind as NU. The value of X shall not be negative.
N	shall be of type integer and nonnegative.

Result Characteristics. Same type and kind as X.

Case (i): The result of CYL_NEUMANN (NU, X) is scalar.

Case (ii): The result of CYL_NEUMANN (NU, X, N) is a rank-one array with extent N.

Result Value.

Case (i): The value of the result of CYL_NEUMANN (NU, X) is a processor-dependent approximation to the Neumann function $N_\nu(x)$ of order NU with argument X, defined by

$$N_\nu(x) = \lim_{\mu \to \nu} \frac{J_\mu(x) \cos \mu x - J_{-\mu}(x)}{\sin \mu x}, \quad x \geq 0$$

and several other representations that appear in the references (1.6, 1.7).

Case (ii): The value of the i'th element of the result of CYL_NEUMANN (NU, X, N) is the same as the value of the result of CYL_NEUMANN (NU+$i-1$, X)

NOTE 2.3

The Neumann function is also known as the cylindrical Bessel function of the second kind, $Y_\nu(x)$.

NOTE 2.4

This is a generalization of the standard intrinsic function BESSEL_YN to noninteger order.

Example. CYL_NEUMANN (0.0, 0.8935769663) has the value 0.0 (approximately).

2.4.9 ELLINT_1 (K) or ELLINT_1 (K, PHI)

Description. Elliptic integral of the first kind.

Class. Elemental function.

Arguments.

K	shall be of type real. The absolute value of K shall be less than or equal to 1.0.
PHI	shall be of type real and the same kind as K.

Result Characteristics. Same as K.

Result Value. The value of the result of ELLINT_1 (K) is a processor-dependent approximation to the complete elliptic integral of the first kind $K(k)$ with argument K, defined by

$$K(k) = \int_0^{\pi/2} \frac{\mathrm{d}\theta}{\sqrt{1 - k^2 \sin^2 \theta}} = \int_0^1 \frac{\mathrm{d}t}{\sqrt{1-t^2}\sqrt{1-k^2t^2}}, \quad |k| \leq 1$$

and several other representations that appear in the references (1.6, 1.7).

The value of the result of ELLINT_1 (K,PHI) is a processor-dependent approximation to the incomplete elliptic integral of the first kind $F(k,\phi)$ with arguments K and PHI, defined by

$$F(k,\phi) = \int_0^{\phi} \frac{\mathrm{d}\theta}{\sqrt{1 - k^2 \sin^2 \theta}}, \quad |k| \leq 1$$

and several other representations that appear in the references (1.6, 1.7).

Examples. ELLINT_1 (0.0) has the value 1.5707963 (approximately). ELLINT_1 (0.0, 1.5707963) has the value 1.5707963 (approximately).

2.4.10 ELLINT_2 (K) or ELLINT_2 (K, PHI)

Description. Elliptic integral of the second kind.

Class. Elemental function.

Arguments.

K	shall be of type real. The absolute value of K shall be less than or equal to 1.0.
PHI	shall be of type real and the same kind as K.

Result Characteristics. Same as K.

Result Value. The value of the result of ELLINT_2 (K) is a processor-dependent approximation to the complete elliptic integral of the second kind $E(k)$ with argument K, defined by

$$E(k) = \int_0^{\pi/2} \sqrt{1 - k^2 \sin^2 \theta}\, \mathrm{d}\theta = \int_0^1 \sqrt{\frac{1 - k^2 t^2}{1 - t^2}} \mathrm{d}t, \quad |k| \le 1$$

and several other representations that appear in the references (1.6, 1.7).

The value of the result of ELLINT_2 (K, PHI) is a processor-dependent approximation to the incomplete elliptic integral of the second kind $E(k, \phi)$ with arguments K and PHI, defined by

$$E(k, \phi) = \int_0^{\phi} \sqrt{1 - k^2 \sin^2 \theta}\, \mathrm{d}\theta, \quad |k| \le 1$$

and several other representations that appear in the references (1.6, 1.7).

Examples. ELLINT_2 (1.0) has the value 1.0 (approximately). ELLINT_2 (1.0, 1.5707963) has the value 1.0 (approximately).

2.4.11 ELLINT_3 (NU, K) or ELLINT_3 (NU, K, PHI)

Description. Elliptic integral of the third kind.

Class. Elemental function.

Arguments.

NU	shall be of type real.
K	shall be of type real and the same kind as NU. The absolute value of K shall be less than or equal to 1.0.
PHI	shall be of type real and the same kind as K.

Result Characteristics. Same as NU.

Result Value. The value of the result of ELLINT_3 (NU, K) is a processor-dependent approximation to the complete elliptic integral of the third kind $\Pi(\nu; k)$ with arguments NU and K, defined by

$$\Pi(\nu; k) = \int_0^{\pi/2} \frac{\mathrm{d}\theta}{(1 + \nu \sin^2 \theta)\sqrt{1 - k^2 \sin^2 \theta}} = \int_0^1 \frac{\mathrm{d}t}{(1 + \nu t^2)\sqrt{(1 - t^2)(1 - k^2 t^2)}}, \quad |k| \le 1$$

and several other representations that appear in the references (1.6, 1.7).

The value of the result of ELLINT_3 (NU, K, PHI) is a processor-dependent approximation to the incomplete elliptic integral of the third kind $\Pi(\nu; k, \phi)$ with arguments NU, K, and PHI, defined by

$$\Pi(\nu; k, \phi) = \int_0^\phi \frac{\mathrm{d}\theta}{(1 + \nu \sin^2\theta)\sqrt{1 - k^2 \sin^2\theta}}, \quad |k| \le 1$$

and several other representations that appear in the references (1.6, 1.7).

Examples. ELLINT_3 (0.0, 0.0) has the value 1.5707963 (approximately). ELLINT_3 (0.0, 0.0, 1.5707963) has the value 1.5707963 (approximately).

2.4.12 EIN (X)

Description. Entire exponential integral.

Class. Elemental function.

Arguments.

X shall be of type real.

Result Characteristics. Same as X.

Result Value. The value of the result is a processor-dependent approximation to the entire exponential integral $\mathrm{Ein}(x)$ with argument X, defined by

$$\mathrm{Ein}(x) = \int_0^x \frac{1 - \exp(-t)}{t} \mathrm{d}t = \gamma + \ln|x| + \mathrm{E}_1(x) = \gamma + \ln|x| - \Re\,\mathrm{Ei}(-x),$$

where $\gamma \approx 0.57721\ 56649$ is Euler's constant, and several other representations that appear in the references (1.6, 1.7).

Example. EIN (1.0) has the value 0.7965995993 (approximately).

2.4.13 EXPINT (X)

Description. Exponential integral.

Class. Elemental function.

Arguments.

X shall be of type real.

Result Characteristics. Same as X.

Result Value. The value of the result is a processor-dependent approximation to the exponential integral $\mathrm{Ei}(x)$ with argument X, defined by

$$\mathrm{Ei}(x) = \int_{-\infty}^x \frac{\exp(t)}{t} \mathrm{d}t = -\int_{-x}^\infty \frac{\exp(-t)}{t} \mathrm{d}t.$$

The integrand is singular at $x = 0$, so for $x > 0$ the integral is interpeted as the Cauchy limit

$$\mathrm{Ei}(x) = \lim_{\epsilon \to 0+} \left(\mathrm{Ei}(-\epsilon) + \int_\epsilon^x \frac{\exp(t)}{t} \mathrm{d}t \right), \quad x > 0.$$

Several other representations appear in the references (1.6, 1.7).

Example. EXPINT (1.0) has the value 1.895117816 (approximately).

2.4.14 HERMITE (N, X)

Description. Hermite polynomial.

Class. Elemental function.

Arguments.

N	shall be of type integer and nonnegative.
X	shall be of type real.

Result Characteristics. Same as X.

Result Value. The value of the result is a processor-dependent approximation to the Hermite polynomial $H_n(x)$ of order N with argument X, defined by the Rodrigues formula

$$H_n(x) = (-1)^n \exp(x^2) \frac{\mathrm{d}^n}{\mathrm{d}x^n} \exp(-x^2),$$

and several other representations that appear in the references (1.6, 1.7).

Example. HERMITE (1, 1.0) has the value 2.0 (approximately).

2.4.15 LAGUERRE (N, X)

Description. Laguerre polynomial.

Class. Elemental function.

Arguments.

N	shall be of type integer and nonnegative.
X	shall be of type real. The value of X shall not be negative.

Result Characteristics. Same as X.

Result Value. The value of the result is a processor-dependent approximation to the Laguerre polynomial $L_n(x)$ of order N with argument X, defined by the Rodrigues formula

$$L_n(x) = \frac{\exp(x)}{n!} \frac{\mathrm{d}^n}{\mathrm{d}x^n} \left(x^n \exp(-x)\right), \quad x \geq 0,$$

and several other representations that appear in the references (1.6, 1.7).

Example. LAGUERRE (1, 1.0) has the value 0.0 (approximately).

2.4.16 LEGENDRE (N, X)

Description. Legendre polynomial.

Class. Elemental function.

Arguments.

N	shall be of type integer and nonnegative.
X	shall be of type real.

Result Characteristics. Same as X.

Result Value. The value of the result is a processor-dependent approximation to the Legendre polynomial $P_n(x)$ of order N with argument X, defined by the Rodrigues formula

$$P_n(x) = \frac{1}{2^n n!} \frac{\mathrm{d}^n}{\mathrm{d}x^n} \left(x^2 - 1\right)^n ,$$

and several other representations that appear in the references (1.6, 1.7).

Example. LEGENDRE (1, 1.0) has the value 1.0 (approximately).

2.4.17 RIEMANN_ZETA (X)

Description. Riemann zeta function.

Class. Elemental function.

Arguments.

X shall be of type real.

Result Characteristics. Same as X.

Result Value. The value of the result is a processor-dependent approximation to the Riemann zeta function $\zeta(x)$ with argument X, defined by

$$\zeta(x) = \begin{cases} \sum_{k=1}^{\infty} k^{-x} & x > 1 \\ \frac{1}{1-2^{1-x}} \sum_{k=1}^{\infty} (-1)^{k-1} k^{-x} & 0 \le x \le 1 \\ 2^x \pi^{x-1} \sin\left(\frac{\pi x}{2}\right) \Gamma(1-x) \zeta(1-x) & x < 0 \end{cases} ,$$

and several other representations that appear in the references (1.6, 1.7).

Example. RIEMANN_ZETA (0.5) has the value −1.460354509 (approximately).

2.4.18 SPH_BESSEL (N, X) or SPH_BESSEL (N1, N2, X)

Description. Spherical Bessel function of the first kind.

Class.

Case (i): SPH_BESSEL (N, X) is an elemental function.

Case (ii): SPH_BESSEL (N1, N2, X) is a transformational function.

Arguments.

N shall be of type integer and nonnegative.

N1 shall be of type integer and nonnegative.

N2 shall be of type integer and nonnegative.

X shall be of type real.

Result Characteristics. Same type and kind as X.

Case (i): The result of SPH_BESSEL (N, X) is scalar.

Case (ii): The result of SPH_BESSEL (N1, N2, X) is a rank-one array with extent MAX(N2-N1+1,0).

Result Value.

Case (i): The value of the result of SPH_BESSEL (N, X) is a processor-dependent approximation to the Spherical Bessel function of the first kind $j_n(x)$ of order N with argument X, defined by

$$j_n(x) = \sqrt{\frac{\pi}{2x}} J_{n+1/2}(x), \quad x \geq 0,$$

where $J_{n+1/2}$ is the cylindrical Bessel function. Several other representations appear in the references (1.6, 1.7).

Case (ii): The value of the i'th element of the result of SPH_BESSEL (N1, N2, X) is the same as the value of SPH_BESSEL (N1+$i-1$, X)

Example. SPH_BESSEL (0, 1.0) has the value 0.8414709848 (approximately).

2.4.19 SPH_LEGENDRE (L, M, THETA)

Description. Spherical associated Legendre function.

Class. Elemental function.

Arguments.

L shall be of type integer.

M shall be of type integer. The absolute value of M shall be less than or equal to the value of L.

THETA shall be of type real.

Result Characteristics. Same as X.

Result Value. The value of the result is a processor-dependent approximation to the Spherical associated Legendre function $Y_\ell^m(\theta, 0)$ of order M and L with argument THETA, where $Y_\ell^m(\theta, \phi)$ is defined by

$$Y_\ell^m(\theta, \phi) = (-1)^m \left[\frac{(2\ell+1)}{4\pi} \frac{(\ell-m)!}{(\ell+m)!} \right]^{1/2} P_\ell^m(\cos\theta) \exp(im\phi), \quad |m| \leq \ell,$$

and several other representations that appear in the references (1.6, 1.7).

NOTE 2.5

Spherical associated Legendre functions are also known as spherical harmonics.

Example. SPH_LEGENDRE (0, 0, 0.0) has the value 0.0 (approximately).

2.4.20 SPH_NEUMANN (N, X) or SPH_NEUMANN (N1, N2, X)

Description. Spherical Neumann function.

Class.

Case (i): SPH_NEUMANN (N, X) is an elemental function.

Case (ii): SPH_NEUMANN (N1, N2, X) is a transformational function.

Arguments.

N shall be of type integer and nonnegative.

N1 shall be of type integer and nonnegative.

N2 shall be of type integer and nonnegative.

X shall be of type real.

Result Characteristics. Same type and kind as X.

Case (i): The result of SPH_NEUMANN (N, X) is scalar.

Case (ii): The result of SPH_NEUMANN (N1, N2, X) is a rank-one array with extent MAX(N2-N1+1,0).

Result Value.

Case (i): The value of the result of SPH_NEUMANN (N, X) is a processor-dependent approximation to the Spherical Neumann function $n_n(x)$ of order N with argument X, defined by

$$n_n(x) = \sqrt{\frac{\pi}{2x}} N_{n+1/2}(x), \quad x \geq 0,$$

where $N_{n+1/2}$ is the Neumann function. Several other representations appear in the references (1.6, 1.7).

Case (ii): The value of the i'th element of the result of SPH_NEUMANN (N1, N2, X) is the same as the value of SPH_NEUMANN (N1+$i-1$, X)

Example. SPH_NEUMANN (1, 1.0) has the value -1.381773291 (approximately).

2.5 Proposed additional procedures

2.5.1 General

Procedures to evaluate the functions proposed in subclause 2.5 do not appear in ISO 31-11:1992, ISO/IEC 24747:2009, or ISO/IEC 29124:2010. Their use appears sufficiently frequently in practice that their availability would be useful.

2.5.2 CHEBYSHEV (N, X) or CHEBYSHEV (N1, N2, X)

Description. Chebyshev polynomial.

Class.

Case (i): CHEBYSHEV (N, X) is an elemental function.

Case (ii): CHEBYSHEV (N1, N2, X) is a transformational function.

Arguments.

N shall be of type integer and nonnegative.

N1 shall be of type integer and nonnegative.

N2 shall be of type integer and nonnegative.

X shall be of type real.

Result Characteristics. Same type and kind as X.

Case (i): The result of CHEBYSHEV (N, X) is a scalar.

Case (ii): The result of CHEBYSHEV (N1, N2, X) is a rank-one array with extent max(N2-N1+1,0).

Result Value.

Case (i): The value of the result of CHEBYSHEV (N, X) is a processor-dependent approximation to the Chebyshev polynomial $T_n(x)$ of order N with argument X, defined by

$$T_n(x) = \cos(n \cos^{-1} x)$$

and other representations that appear in the references (1.6, 1.7).

Case (ii): The i'th element of the value of the result of CHEBYSHEV (N1, N2, X) is the same as the value of CHEBYSHEV (N1+$i-1$, X).

Example. CHEBYSEV (1, 1.0) has the value 1.0 (approximately).

2.5.3 CI (X)

Description. Cosine integral.

Class. Elemental function.

Arguments.

X shall be of type real. The value of X shall not be zero.

Result Characteristics. Same as X.

Result Value. The value of the result is a processor-dependent approximation to the cosine integral $\mathrm{Ci}(x)$ with argument X, defined by

$$\mathrm{Ci}(x) = -\int_x^\infty \frac{\cos t}{t} \mathrm{d}t = \gamma + \ln|x| - \mathrm{Cin}(x),$$

and other representations that appear in the references (1.6, 1.7).

The integrand is singular at zero, so for $x < 0$ the integral is interpeted as the Cauchy limit

$$\mathrm{Ci}(x) = \lim_{\epsilon \to 0-} \left(\int_x^\epsilon \frac{\cos(t)}{t} \mathrm{d}t + \mathrm{Ci}(-x) \right)$$

Example. CI (1.0) has the value 0.3374039229 (approximately).

2.5.4 CIN (X)

Description. Entire cosine integral.

Class. Elemental function.

Arguments.

X shall be of type real.

Result Characteristics. Same as X.

Result Value. The value of the result is a processor-dependent approximation to the entire cosine integral $\mathrm{Cin}(x)$ with argument X, defined by

$$\mathrm{Cin}(x) = \int_0^x \frac{1-\cos t}{t} \mathrm{d}t = \gamma + \ln|x| - \mathrm{Ci}(x),$$

and other representations that appear in the references (1.6, 1.7).

Example. CIN (1.0) has the value 0.2398117420 (approximately).

2.5.5 DAW (X)

Description. Dawson function.

Class. Elemental function.

Arguments.

X shall be of type real.

Result Characteristics. Same as X.

Result Value. The value of the result is a processor-dependent approximation to the Dawson function daw(x) with argument X, defined by

$$\mathrm{daw}(x) = \int_0^x \exp(t^2 - x^2)\,\mathrm{d}t$$

and other representations that appear in the references (1.6, 1.7).

Example. DAW (1.0) has the value 0.5380795069 (approximately).

2.5.6 ERFCI (X)

Description. Inverse co-error function.

Class. Elemental function.

Arguments.

X shall be of type real.

Result Characteristics. Same as X.

Result Value. The value of the result is a processor-dependent approximation to the value y such that $x = \mathrm{erfc}(y)$, that is

$$x = \mathrm{erfc}(y) = \frac{2}{\sqrt{\pi}} \int_y^\infty \exp(-t^2)\,\mathrm{d}t,$$

and other representations that appear in the references (1.6, 1.7).

Example. ERFCI (0.5) has the value 0.5230637238 (approximately).

2.5.7 ERFI (X)

Description. Inverse error function.

Class. Elemental function.

Arguments.

X shall be of type real.

Result Characteristics. Same as X.

Result Value. The value of the result is a processor-dependent approximation to the value y such that $x = \mathrm{erf}(y)$, that is

$$x = \mathrm{erf}(y) = \frac{2}{\sqrt{\pi}} \int_0^y \exp(-t^2)\,\mathrm{d}t,$$

and other representations that appear in the references (1.6, 1.7).

Example. ERFI (0.5) has the value 0.4769362762 (approximately).

2.5.8 FRESNEL_C (X)

Description. Fresnel cosine integral.

Class. Elemental function.

Arguments.

X shall be of type real.

Result Characteristics. Same as X.

Result Value. The value of the result is a processor-dependent approximation to the Fresnel cosine integral C(x) with argument X, defined by

$$\mathrm{C}(x) = \int_0^x \cos\left(\frac{\pi}{2}t^2\right)\, \mathrm{d}t,$$

and other representations that appear in the references (1.6, 1.7).

Example. FRESNEL_C (1.0) has the value 0.7798934004 (approximately).

2.5.9 FRESNEL_S (X)

Description. Fresnel sine integral.

Class. Elemental function.

Arguments.

X shall be of type real.

Result Characteristics. Same as X.

Result Value. The value of the result is a processor-dependent approximation to the Fresnel sine integral S(x) with argument X, defined by

$$\mathrm{S}(x) = \int_0^x \sin\left(\frac{\pi}{2}t^2\right)\, \mathrm{d}t,$$

and other representations that appear in the references (1.6, 1.7).

Example. FRESNEL_S (1.0) has the value 0.4382591474 (approximately).

2.5.10 INCOMPLETE_GAMMA_RATIOS (NU, X, P, Q)

Description. Incomplete gamma function ratios.

Class. Elemental subroutine.

Arguments.

NU shall be of type real. The value of NU shall be greater than zero. NU is an INTENT(IN) argument.

X shall be of type real and the same kind as NU. The value of X shall be greater than zero. X is an INTENT(IN) argument.

P shall be of type real and the same kind as NU. P is an INTENT(OUT) argument.

Q shall be of type real and the same kind as NU. Q is an INTENT(OUT) argument.

Result Value. The values of the P and Q arguments are processor-dependent approximations to the incomplete gamma ratios $P(\nu;x)$ and $Q(\nu;x)$ with arguments NU and X, defined by

$$P(\nu;x) = \frac{\gamma(\nu;x)}{\Gamma(\nu)} \text{ and } Q(\nu;x) = \frac{\Gamma(\nu;x)}{\Gamma(\nu)}, \text{ where}$$

$$\gamma(\nu;x) = \int_0^x t^{\nu-1} \exp(-t)\,\mathrm{d}t \text{ and } \Gamma(\nu;x) = \int_x^\infty t^{\nu-1} \exp(-t)\,\mathrm{d}t, \quad x > 0,\ \nu > 0,$$

and other representations that appear in the references (1.6, 1.7).

Example. After executing CALL INCOMPLETE_GAMMA_RATIO (1.0, 1.0, P, Q), the variables P and Q have the values 0.6321205588 and 0.3678794412, respectively (approximately).

NOTE 2.6

$P(\nu;x)+Q(\nu;x) = 1$, but they are not equally well conditioned computationally. In general, when one is small, it should not be computed by subtracting the other from 1.0. When $\nu \approx x$ and $\nu >> 0$, $x >> 0$ they are both very poorly conditioned.

2.5.11 SI (X)

Description. Sine integral.

Class. Elemental function.

Arguments.

X shall be of type real.

Result Characteristics. Same as X.

Result Value. The value of the result is a processor-dependent approximation to the sine integral $\mathrm{Si}(x)$, defined by

$$\mathrm{Si}(x) = \int_0^x \frac{\sin t}{t}\mathrm{d}t,$$

and other representations that appear in the references (1.6, 1.7).

Example. SI (1.0) has the value 0.9460830704 (approximately).

Annex A

(Informative)

Processor dependencies

According to this part of ISO/IEC 1539, the following are processor dependent:

- Whether a processor supports this part of ISO/IEC 1539 (1.1).
- Whether a processor provides specific procedures for arguments of type integer and kind other than default integer kind (2.1).
- Whether the module procedures are separate procedures, and if so, the submodule identifiers of the submodules in which the procedures are defined (2.1).
- The kind and radix of the named constant `EULER_GAMMA` (2.2.1).
- The mathematical approximations used (2.4, 2.5).