

Mathematical Special Functions for C++17, v5

Document #: WG21 P0226R1
Date: 2016-02-29
Revises: [N4437](#), [N3743](#), [N3548](#)
Project: JTC1.22.32 Programming Language C++
Audience: LEWG \Rightarrow LWG
Reply to: Walter E. Brown <webrown.cpp@gmail.com>
Axel Naumann <Axel.Naumann@cern.ch>
Edward Smith-Rowland <ESmith-rowland@alionscience.com>

Contents

| | | | | | |
|---|-----------------------------------|----------|---|----------------------------|----------|
| 1 | Background and proposal | 1 | 4 | Acknowledgments | 3 |
| 2 | Proposed wording | 2 | 5 | Bibliography | 3 |
| 3 | Macros | 3 | 6 | Document history | 5 |

Abstract

This paper proposes to merge International Standard 29124:2010, “Extensions to the C++ Library to support mathematical special functions,” into the C++17 Working Paper’s specification of the `<cmath>` header.

1 Background and proposal

Mathematical Special Functions were proposed [[N1422](#), [N1514](#), [N1542](#)] for C++ with the following rationale:

While these functions are clearly numerical in nature and will likely be most heavily used by the scientific and engineering communities, other communities of programmers also have needs, ranging from frequent to intermittent, for these functions. . . . [W]e believe that adoption of this proposal would send a clear message to the various numeric computing communities that, contrary to significant popular belief within these communities, C++ is an eminently suitable programming language for their problem domain, too.

Modulo some additional discussion and wordsmithing [[N1502](#), [N1570](#), [N1665](#), [N1884](#)], the proposal’s wording was subsequently published as one of the clauses comprising WG21’s TR1 [[ISO07](#)].

When it came time to determine what to do with TR1 relative to C++0X, WG21 decided¹ to adopt essentially all of TR1 into the Working Paper, with the notable exclusion of the clause specifying the Mathematical Special Functions. During the discussion that led to that decision, it was principally argued that these functions’ implementation would be a development hardship and maintenance burden for vendors, noting that “Users aren’t breaking down our doors clamoring

Copyright © 2016 by Walter E. Brown. All rights reserved.

¹The decision was made at the Berlin meeting (2006) via two motions.

for these functions."² Still later, following WG14's lead [ISO09],³ WG21 voted to issue its own Mathematical Special Functions International Standard, IS 29124:2010 [N3060, ISO10].⁴

With encouragement/approval from Study Group 6 (numerics) at previous WG21 meetings, [N4437] proposed to include IS 29124:2010 in the C++ standard as a conditionally-supported feature. Upon reflection and (lively!) discussion in Lenexa, WG21 recommended revision so as to *require* that IS 29124:2010's functionality be made unconditionally available. This recommendation was based on the following observations:

- that several standard library vendors (Dinkumware, Boost, gcc) have complete implementations of the IS,
- that WG21 preferred to avoid including a conditionally supported library feature in the standard, and
- (most significantly) that a conditionally supported feature could not be relied upon across implementations or platforms, rendering it useless for certain user communities.

We therefore **propose to merge IS 29124:2010 [ISO10] into the C++17 standard as a Standard Library feature in header `<cmath>`**. We do so now for the following principal reasons:

- “There is a long history of implementation experience with these functions. . . . All the functions are mathematically well-understood, all have proven their utility in practice over a considerable period of time, and all have been previously implemented in C and C++” [N1422, N1514, N1542];
- The merger will decrease future WG21 burden by managing a single International Standard, rather than two. This reduces not only the number of published Standards, but also the amount of such future overhead as working papers, issues lists, Project Editors, and the like.
- More importantly, members of the user community have expressed interest in maintaining and enhancing the functionality provided by IS 29124:2010. (See, for example, [N3494].) However, WG21 has no vehicle to do so, as IS 29124:2010 is considered a completed project. If it were merged into the C++17 standard, our usual process would be able to evaluate and act on such proposals.
- Finally, it is relevant to take this step in time for C++17, because IS 29124:2010 is now (past) due per the usual ISO rule for periodic review. This is thus an appropriate time to plan this Standard's retirement.

2 Proposed wording

All proposed wording is relative to WG21 draft [N4567]. Text [like this](#) is to be added, text ~~like this~~ is to be deleted, and editorial notes are displayed • like this.⁵

²However, it was evidently not deemed a similar hardship and burden to integrate C99's then-new math functions (some of which, like the hyperbolic functions, are actually mathematically classified as special functions) into C++11. As enumerated by P. J. Plauger in [N1372], the functions added to `<math.h>` consisted of “`acosh`, `asinh`, `atanh`, `cbirt`, `copysign`, `erf`, `erfc`, `exp2`, `expm1`, `fdim`, `fma`, `fmax`, `fmin`, `hypot`, `ilogb`, `lgamma`, `llrint`, `llround`, `log1p`, `log2`, `logb`, `lrint`, `lround`, `nan`, `nearbyint`, `nextafter`, `nexttoward`, `remainder`, `remquo`, `rint`, `round`, `scalbln`, `scanbn`, `tgamma`, and `trunc`, plus float versions (ending in `f`) and long double versions (ending in `l`)” as well as additional new macros, function macros, and entirely new related headers such as `<fenv.h>`. ☺

³In explaining why it wanted to issue an International Standard rather than a Technical Report on this topic, John Benito noted on behalf of WG14 that the state of the art had advanced considerably since the project's inception: “[Now t]here are several implementations, complete or under development, all using the functionality that the committee intended to document. . . .” [Ben07].

⁴According to WG21 Convener Herb Sutter, these two International Standards are reportedly the first in ISO history that were approved after two ballots each, rather than via the then-usual three-ballot process, because there were no “no” votes.

⁵The technical content of [ISO10], in the form of \LaTeX machine-readable text, will be made available to the WG21 Project Editor to help insure the smoothest possible transition.

- At the level of and following [c.math], create a new subclause with heading and initial content the same as IS 29124:2010’s clause [sf.cmath], “Additions to header <cmath>,” renumbering as appropriate to the new context. Then adjust the newly-imported [sf.cmath]/2 as follows:

... The detailed signatures added in namespace std to header <cmath> are: ...

- Clarifying that the mathematical special functions are intended not to be added also to the (deprecated) <math.h> header, adjust [depr.c.headers]/2 as follows, replacing N with the correct subclause number so as to refer to the newly-added text above:

Every Except for the functions declared in clause 26.N, every C header, each of which has a name of the form **name.h**, behaves as if each name placed in the standard library namespace by the corresponding *cname* header is placed within the global namespace scope. ...

- Merge IS 29124:2010’s clause [norm], “Normative References,” into [intro.refs], and merge IS 29124:2010’s clause [terms], “Terms, definitions, and symbols,” into [intro.defs], each in an order determined at the discretion of the Project Editor.

3 Macros

Since IS 24747:2009 was designed as an opt-in library (what we would today term “conditionally-supported”), it furnished its own macro-based mechanisms for that purpose. This proposal does not preserve such macros.

However, for the purposes of SG10, we recommend a feature test macro by the name `__cpp_lib_math_special_functions`. While the IS does have a macro intended for an analogous purpose, it predates and hence does not follow SG10’s recommended naming conventions. We recommend abandoning the IS’s now-dated identification and enabling macros.

4 Acknowledgments

Many thanks to everyone who helped during the evolution of Mathematical Special Functions in the C++ standard library.

Special thanks to WG14 for its early and enthusiastic embrace of Mathematical Special Functions for the C library.

5 Bibliography

- [Ben07] [John] Benito: “Rationale to change ISO/IEC JTC 1/SC 22 Program of Work for TR 24727 to IS 24747.” ISO/IEC JTC1/SC22/WG14 document N1234, 2007-04-28.
<http://www.open-std.org/jtc1/sc22/wg14/www/docs/n1234.htm>.
- [ISO07] International Organization for Standardization: “Information technology — Programming languages — Technical Report on C++ Library Extensions.” ISO/IEC document TR 19768:2007.
- [ISO09] International Organization for Standardization: “Information technology — Programming languages — Programming languages, their environments and system software interfaces — Extensions to the C Library to support mathematical special functions.” ISO/IEC document IS 24747:2009.

- [ISO10] International Organization for Standardization: "Information technology — Programming languages — Programming languages, their environments and system software interfaces — Extensions to the C++ Library to support mathematical special functions." ISO/IEC document IS 29124:2010.
- [N1372] P. J. Plauger: "Proposed C99 Library Additions to C++ (Revised)." ISO/IEC JTC1/SC22/WG21 document N1372 (post-Curacao mailing), Undated.
<http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2002/n1372.txt>.
- [N1422] Walter E. Brown: "A Proposal to Add Mathematical Special Functions to the C++ Standard Library." ISO/IEC JTC1/SC22/WG21 document N1422 (pre-Oxford mailing), 2003-02-24.
<http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2003/n1422.html>.
- [N1502] P. J. Plauger: "Proposed Signature Changes for Special Math Functions in TR-1." ISO/IEC JTC1/SC22/WG21 document N1502 (pre-Kona mailing), Undated.
<http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2003/n1502.txt>.
- [N1514] Walter E. Brown: "A Proposal to Add Mathematical Special Functions to the C++ Standard Library (version 2)." ISO/IEC JTC1/SC22/WG21 document N1514 (pre-Kona mailing), 2003-05-15.
<http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2003/n1514.pdf>.
- [N1542] Walter E. Brown: "A Proposal to Add Mathematical Special Functions to the C++ Standard Library (version 3)." ISO/IEC JTC1/SC22/WG21 document N1542 (post-Kona mailing), 2003-10-28.
<http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2003/n1542.pdf>.
- [N1564] William M. Miller: "Core Issue 195 and 'Conditionally-Supported Behavior'." ISO/IEC JTC1/SC22/WG21 document N1564 (pre-Sydney mailing), 2004-02-12.
<http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2004/n1564.pdf>.
- [N1570] P. J. Plauger: "Corrections to Domain-error Reporting for TR1 Chapter on Special Math Functions." ISO/IEC JTC1/SC22/WG21 document N1570 (pre-Sydney mailing), Undated.
<http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2004/n1570.txt>.
- [N1627] William M. Miller: "'Conditionally-Supported Behavior' (Rev. 1)." ISO/IEC JTC1/SC22/WG21 document N1627 (post-Sydney mailing), 2004-04-04.
<http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2004/n1627.pdf>.
- [N1665] John Marraffino and Marc Paterno: "Guidelines for Domain Errors in Mathematical Special Functions." ISO/IEC JTC1/SC22/WG21 document N1665 (mid-Sydney/Redmond mailing), 2004-07-15.
<http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2004/n1665.pdf>.
- [N1884] P. J. Plauger and Chris Walker: "Further Restrictions on Special Math Functions." ISO/IEC JTC1/SC22/WG21 document N1884 (pre-Tremblant mailing), 2005-08-26.
<http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2005/n1884.txt>.
- [N3060] Walter E. Brown: "Final Draft International Standard 29124." ISO/IEC JTC1/SC22/WG21 document N3060 (post-Pittsburgh mailing), 2010-03-06.
<http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2010/n3060.pdf>.
- [N3061] Walter E. Brown: "Record of Response: ISO/IEC 29124." ISO/IEC JTC1/SC22/WG21 document N3061 (post-Pittsburgh mailing), 2010-03-06.
<http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2010/n3061.pdf>.
- [N3494] Vincent Reverdý: "A Proposal to Add Special Mathematical Functions According to the ISO/IEC 80000-2:2009 standard." ISO/IEC JTC1/SC22/WG21 document N3494 (mid-Portland/Bristol mailing), 2012-12-19.
<http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2012/n3494.pdf>.
- [N4437] Walter E. Brown: "Conditionally-supported Special Math Functions, v3." ISO/IEC JTC1/SC22/WG21 document N4437 (pre-Lenexa mailing), 2015-04-09.
<http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2015/n4437.pdf>.
- [N4567] Richard Smith: "Working Draft, Standard for Programming Language C++." ISO/IEC JTC1/SC22/WG21 document N4567 (post-Kona mailing), 2015-11-09.
<http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2015/n4567.pdf>.

6 Document history

| Version | Date | Changes |
|---------|------------|--|
| 1 | 2013-03-12 | • Published as N3548. |
| 2 | 2013-08-30 | • Removed inadvertently duplicated word. • Corrected N1884's authorship; apologies for the accidental omission. • Identified the WG21 Convener by name as well as title. • Updated relative to the latest Working Draft. • Published as N3743. |
| 3 | 2015-04-09 | • Updated to propose C++17 incorporation (rather than C++14). • Updated relative to the latest Working Draft. • Published as N4437. |
| 4 | 2016-02-10 | • Summarized WG21's views re N4437. • Removed "conditionally-supported" parts and revised title/abstract/proposed wording accordingly. • Identified vendors who have implementations. • Updated relative to the latest Working Draft. • Clarified that <code><cmath></code> (not <code><math.h></code>) is affected. • Added §3 (macros). • Published as P0226R0. |
| 5 | 2016-02-29 | • Added wording for subclause [depr.c.headers] per LEWG @ Jacksonville. • Published as P0226R1. |