

ISO/IEC TR 24733:2011-11 (E)

Information technology - Programming languages, their environments and system software interfaces - Extensions for the programming language C++ to support decimal floating-point arithmetic

Contents		Page
Foreword		vi
Introduction		vii
0.1	General	vii
0.2	Arithmetic model	vii
0.3	The Formats	viii
1	Scope	1
2	Normative references	1
3	Conventions	2
3.1	General	2
3.2	Relation to C++ Standard Library Introduction	2
3.3	Relation to "Technical Report on C++ Library Extensions"	3
3.4	Categories of extensions	3
3.5	Namespaces and headers	4
4	Decimal floating-point types	5
4.1	Characteristics of decimal floating-point types	5
4.2	Decimal Types	6
4.2.1	Header <decimal> synopsis	6
4.2.2	Class decimal32	10
4.2.2.1	Class summary	10
4.2.2.2	Construct/copy/destroy	11
4.2.2.3	Conversion from floating-point type	11
4.2.2.4	Conversion from integral type	11
4.2.2.5	Conversion to integral type	12
4.2.2.6	Increment and decrement operators	12
4.2.2.7	Compound assignment	12
4.2.3	Class decimal64	14
4.2.3.1	Class summary	14
4.2.3.2	Construct/copy/destroy	14
4.2.3.3	Conversion from floating-point type	14
4.2.3.4	Conversion from integral type	15
4.2.3.5	Conversion to integral type	15
4.2.3.6	Increment and decrement operators	15
4.2.3.7	Compound assignment	16
4.2.4	Class decimal128	17
4.2.4.1	Class summary	17
4.2.4.2	Construct/copy/destroy	17
4.2.4.3	Conversion from floating-point type	17
4.2.4.4	Conversion from integral type	18
4.2.4.5	Conversion to integral type	18
4.2.4.6	Increment and decrement operators	18
4.2.4.7	Compound assignment	19
4.2.5	Initialization from coefficient and exponent	20
4.2.6	Conversion to generic floating-point type	20

4.2.7	Unary arithmetic operators	21
4.2.8	Binary arithmetic operators	21
4.2.9	Comparison operators	22
4.2.10	Formatted input	25
4.2.11	Formatted output	26
4.3	Additions to header <limits>	27
4.4	Headers <cfloat> and <float.h>	30
4.4.1	General	30
4.4.2	Additions to header <float.h> synopsis	30
4.4.3	Additions to header <float.h> synopsis	31
4.4.4	Maximum finite value	31
4.4.5	Epsilon	31
4.4.6	Minimum positive normal value	32
4.4.7	Minimum positive subnormal value	32
4.4.8	Evaluation format	32
4.5	Additions to <cfenv> and <fenv.h>	33
4.5.1	General	33
4.5.2	Additions to <cfenv> synopsis	33
4.5.3	Rounding modes	34
4.5.4	The fe_dec_getround function	34
4.5.5	The fe_dec_setround function	35
4.5.6	Changes to <fenv.h>	35
4.6	Additions to <cmath> and <math.h>	35
4.6.1	General	35
4.6.2	Additions to header <cmath> synopsis	36
4.6.3	<cmath> macros	41
4.6.4	Evaluation formats	41
4.6.5	samequantum functions	42
4.6.6	quantexp functions	42
4.6.7	quantize functions	43
4.6.8	Elementary functions	43
4.6.9	abs function overloads	45
4.6.10	Changes to <math.h>	45
4.6.10.1	General	45
4.6.10.2	Additions to header <math.h> synopsis	45
4.7	Additions to <cstdlib> and <stdio.h>	45
4.8	Additions to <stdlib> and <stdlib.h>	45
4.8.1	Additions to header <stdlib> synopsis	45
4.8.2	strtod functions	45
4.8.3	Changes to <stdlib.h>	45
4.9	Additions to <wchar> and <wchar.h>	46
4.9.1	Additions to <wchar> synopsis	46
4.9.2	wcstod functions	46
4.9.3	Changes to <wchar.h>	46
4.10	Facets	46
4.10.1	General	46
4.10.2	Additions to header <locale> synopsis	47
4.10.3	Class template extended_num_get	47
4.10.3.1	extended_num_get members	49
4.10.3.2	extended_num_get virtual functions	50
4.10.4	Class template extended_num_put	51
4.10.4.1	extended_num_put members	52
4.10.4.2	extended_num_put virtual functions	53
4.11	Type traits	53
4.11.1	Addition to header <type_traits> synopsis	53
4.11.2	is_decimal_floating_point type_trait	54
4.12	Hash functions	54
4.12.1	Additions to header <functional> synopsis	54
4.12.2	Hash function specializations	54
5	Notes on C compatibility	55
5.1	General	55
5.2	Literals	55
5.3	Conversions	55