Accredited Standards Committee*
X3, INFORMATION PROCESSING SYSTEMS

Doc No: X3J16/94−0028
WG21/N0415
Date: 25 January 1994
Project: Programming Language C++
Reply to: Andrew Koenig
AT&T Bell Laboratories
PO Box 636
600 Mountain Avenue
Room 6D−416B
Murray Hill, NJ 07974 USA
ark@research.att.com

Concordance for March 1994 (San Diego) C++ Working Paper

This document provides a cross-refrerence between absolute section numbers and the corresponding symbolic names used for cross-references within the Working Paper. Its purpose is to make it easier to replace absolute section numbers by their corresponding names, which in turn will make it easier to renumber sections in future versions of the Working Paper.

* *Operating under the procedures of the American National Standards Institute (ANSI)*
Standards Secretariat: CBEMA, 1250 Eye Street NW, Suite 200, Washington DC 20005

Listing by clause number

1	intro	General
1.1	intro.scope	Scope
1.2	intro.refs	Normative references
1.3	intro.defs	Definitions
1.4	syntax	Syntax notation
1.5	intro.memory	The C++ memory model
1.6	intro.compliance	Processor compliance
1.7	intro.execution	Program execution
2	lex	Lexical conventions
2.1	lex.phases	Phases of translation
2.2	lex.trigraph	Trigraph sequences
2.3	lex.pptoken	Preprocessing tokens
2.4	lex.digraph	Digraph sequences
2.5	lex.token	Tokens
2.6	lex.comment	Comments
2.7	lex.name	Identifiers
2.8	lex.key	Keywords
2.9	lex.literal	Literals
2.9.1	lex.icon	Integer literals
2.9.2	lex.ccon	Character literals
2.9.3	lex.fcon	Floating literals
2.9.4	lex.string	String literals
2.9.5	lex.bool	Boolean literals
3	basic	Basic concepts
3.1	basic.def	Declarations and definitions
3.2	basic.def.odr	One definition rule
3.3	basic.scope	Declarative regions and scopes
3.3.1	basic.scope.local	Local scope
3.3.2	basic.scope.proto	Function prototype scope
3.3.3		Function scope
3.3.4	basic.file.scope	File scope
3.3.5	basic.scope.namespace	Namespace scope
3.3.6	basic.scope.class	Class scope
3.3.7	basic.scope.hiding	Name hiding
3.3.8	basic.scope.exqual	Explicit qualification
3.3.9	basic.scope.elab	Elaborated type specifier
3.3.10	basic.scope.pdecl	Point of declaration
3.4	basic.link	Program and linkage
3.5	basic.start	Start and termination
3.5.1	basic.start.main	Main function
3.5.2	basic.start.init	Initialization of non-local objects
3.6	basic.start.term	Termination
3.7	basic.stc	Storage duration
3.7.1	basic.stc.static	Static storage duration
3.7.2	basic.stc.auto	Automatic storage duration
3.7.3	basic.stc.dynamic	Dynamic storage class
3.7.4	basic.stc.inherit	Duration of sub-objects
3.7.5	basic.stc.mutable	The `mutable` keyword
3.7.6	basic.stc.ref	Reference duration
3.8	basic.types	Types
3.8.1	basic.fundamental	Fundamental types
3.8.2	basic.compound	Compound types
3.8.3	basic.type.qualifier	CV-qualifiers

3.8.4	basic.type.name	Type names
3.9	basic.lval	Lvalues and rvalues
4	conv	Standard conversions
4.1	conv.prom	Integral promotions
4.2	conv.integral	Integral conversions
4.3	conv.double	Float and double
4.4	conv.float	Floating and integral
4.5	conv.arith	Arithmetic conversions
4.6	conv.ptr	Pointer conversions
4.7	conv.ref	Reference conversions
4.8	conv.mem	Pointers to members
4.9	conv.bool	Boolean conversions
5	expr	Expressions
5.1	expr.prim	Primary expressions
5.2	expr.post	Postfix expressions
5.2.1	expr.sub	Subscripting
5.2.2	expr.call	Function call
5.2.3	expr.type.conv	Explicit type conversion (functional notation)
5.2.4	expr.ref	Class member access
5.2.5	expr.post.incr	Increment and decrement
5.2.6	expr.dynamic.cast	Dynamic cast
5.2.7	expr.typeid	Type identification
5.2.8	expr.static.cast	Static cast
5.2.9	expr.reinterpret.cast	Reinterpret cast
5.2.10	expr.const.cast	Const cast
5.3	expr.unary	Unary expressions
5.3.1	expr.unary.op	Unary operators
5.3.2	expr.pre.incr	Increment and decrement
5.3.3	expr.sizeof	Sizeof
5.3.4	expr.new	New
5.3.5	expr.delete	Delete
5.4	expr.cast	Explicit type conversion (cast notation)
5.5	expr.mptr.oper	Pointer-to-member operators
5.6	expr.mul	Multiplicative operators
5.7	expr.add	Additive operators
5.8	expr.shift	Shift operators
5.9	expr.rel	Relational operators
5.10	expr.eq	Equality operators
5.11	expr.bit.and	Bitwise AND operator
5.12	expr.xor	Bitwise exclusive OR operator
5.13	expr.or	Bitwise inclusive OR operator
5.14	expr.log.and	Logical AND operator
5.15	expr.log.or	Logical OR operator
5.16	expr.cond	Conditional operator
5.17	expr.ass	Assignment operators
5.18	expr.comma	Comma operator
5.19	expr.const	Constant expressions
6	stmt.stmt	Statements
6.1	stmt.label	Labeled statement
6.2	stmt.expr	Expression statement
6.3	stmt.block	Compound statement or block
6.4	stmt.select	Selection statements
6.4.1	stmt.if	The `if` statement
6.4.2	stmt.switch	The `switch` statement
6.5	stmt.iter	Iteration statements

6.5.1	stmt.while	The `while` statement
6.5.2	stmt.do	The `do` statement
6.5.3	stmt.for	The `for` statement
6.6	stmt.jump	Jump statements
6.6.1	stmt.break	The `break` statement
6.6.2	stmt.cont	The `continue` statement
6.6.3	stmt.return	The `return` statement
6.6.4	stmt.goto	The `goto` statement
6.7	stmt.dcl	Declaration statement
6.8	stmt.ambig	Ambiguity resolution
7	dcl.dcl	Declarations
7.1	dcl.spec	Specifiers
7.1.1	dcl.stc	Storage class specifiers
7.1.2	dcl.fct.spec	Function specifiers
7.1.3	dcl.typedef	The `typedef` specifier
7.1.4	dcl.friend	The `friend` specifier
7.1.5	dcl.type	Type specifiers
7.1.5.1	dcl.type.cv	The *cv-qualifiers*
7.1.5.2	dcl.type.simple	Simple type specifiers
7.1.5.3	dcl.type.elab	Elaborated type specifiers
7.2	dcl.enum	Enumeration declarations
7.3	basic.namespace	Namespaces
7.3.1	namespace.def	Namespace definition
7.3.2	namespace.alias	Namespace or class alias
7.3.3	namespace.udecl	The `using` declaration
7.3.4	namespace.udir	Using directive
7.3.5	namespace.qual	Explict qualification
7.4	dcl.asm	The `asm` declaration
7.5	dcl.link	Linkage specifications
8	dcl.decl	Declarators
8.1	dcl.name	Type names
8.2	dcl.ambig.res	Ambiguity resolution
8.3	dcl.meaning	Meaning of declarators
8.3.1	dcl.ptr	Pointers
8.3.2	dcl.ref	References
8.3.3	dcl.mptr	Pointers to members
8.3.4	dcl.array	Arrays
8.3.5	dcl.fct	Functions
8.3.6	dcl.fct.default	Default arguments
8.4	dcl.fct.def	Function definitions
8.5	dcl.init	Initializers
8.5.1	dcl.init.aggr	Aggregates
8.5.2	dcl.init.string	Character arrays
8.5.3	dcl.init.ref	References
9	class	Classes
9.1	class.name	Class names
9.2	class.mem	Class members
9.3	class.scope0	Scope rules for classes
9.4	class.mfct	Member functions
9.4.1	class.this	The `this` pointer
9.4.2	class.inline	Inline member functions
9.5	class.static	Static members
9.6	class.union	Unions
9.7	class.bit	Bit-fields
9.8	class.nest	Nested class declarations

9.9	class.local	Local class declarations
9.10	class.nested.type	Nested type names
10	class.derived	Derived classes
10.1	class.mi	Multiple base classes
10.2	class.ambig	Ambiguities
10.3	class.virtual	Virtual functions
10.4	class.abstract	Abstract classes
10.5	class.scope	Summary of scope rules
11	class.access	Member access control
11.1	class.access.spec	Access specifiers
11.2	class.access.base	Access specifiers for base classes
11.3	class.access.dcl	Access declarations
11.4	class.friend	Friends
11.5	class.protected	Protected member access
11.6	class.access.virt	Access to virtual functions
11.7	class.paths	Multiple access
12	special	Special member functions
12.1	class.ctor	Constructors
12.2	class.temporary	Temporary objects
12.3	class.conv	Conversions
12.3.1	class.conv.ctor	Conversion by constructor
12.3.2	class.conv.fct	Conversion functions
12.4	class.dtor	Destructors
12.5	class.free	Free store
12.6	class.init	Initialization
12.6.1	class.expl.init	Explicit initialization
12.6.2	class.base.init	Initializing bases and members
12.7	class.cdtor	Constructors and destructors
12.8	class.copy	Copying class objects
13	over	Overloading
13.1	over.dcl	Declaration matching
13.2	over.match	Overload resolution
13.2.1	over.match.funcs	Candidate functions
13.2.2	over.match.args	Argument matching
13.3	over.over	Address of overloaded function
13.4	over.oper	Overloaded operators
13.4.1	over.unary	Unary operators
13.4.2	over.binary	Binary operators
13.4.3	over.ass	Assignment
13.4.4	over.call	Function call
13.4.5	over.sub	Subscripting
13.4.6	over.ref	Class member access
13.4.7	over.inc	Increment and decrement
13.4.8	over.oper.funcs	Overloaded operators in expressions
14	temp	Templates
14.1	temp.names	Template names
14.2	temp.res	Name resolution
14.2.1	temp.local	Locally declared names
14.2.2	temp.encl	Names from the template’s enclosing scope
14.2.3	temp.dep	Dependent names
14.2.4	temp.inject	Non-local names declared within a template
14.3	temp.inst	Template instantiation
14.4	temp.explicit	Explicit instantiation
14.5	temp.spec	Template specialization
14.6	temp.param	Template parameters

14.7	temp.arg	Template arguments
14.8	temp.type	Type equivalence
14.9	temp.fct	Function templates
14.9.1	temp.arg.explicit	Explicit template argument specification
14.9.2	temp.deduct	Template argument deduction
14.9.3	temp.over	Overload resolution
14.9.4	temp.over.spec	Overloading and specialization
14.10	temp.mem.func	Member function templates
14.11	temp.friend	Friends
14.12	temp.static	Static members and variables
15	except	Exception handling
15.1	except.intro	Exception handling
15.2	except.throw	Throwing an exception
15.3	except.ctor	Constructors and destructors
15.4	except.handle	Handling an exception
15.5	except.spec	Exception specifications
15.6	except.special	Special functions
15.6.1	except.terminate	The `terminate()` function
15.6.2	except.unexpected	The `unexpected()` function
15.7	except.access	Exceptions and access
16	cpp	Preprocessing directives
16.1	cpp.cond	Conditional inclusion
16.2	cpp.include	Source file inclusion
16.3	cpp.replace	Macro replacement
16.3.1	cpp.subst	Argument substitution
16.3.2	cpp.stringize	The `#` operator
16.3.3	cpp.concat	The `##` operator
16.3.4	cpp.rescan	Rescanning and further replacement
16.3.5	cpp.scope	Scope of macro definitions
16.4	cpp.line	Line control
16.5	cpp.error	Error directive
16.6	cpp.pragma	Pragma directive
16.7	cpp.null	Null directive
16.8	cpp.predefined	Predefined macro names
17	lib.library	Library
17.1	lib.introduction	Introduction
17.1.1	lib.intro.standard.c	Standard C library
17.1.2	lib.headers	Headers
17.1.3	lib.namespaces	Namespaces
17.1.4	lib.reserved.names	Reserved names
17.1.5	lib.res.and.conventions	Restrictions and conventions
17.1.5.1	lib.res.on.macro.definitions	Restrictions on macro definitions
17.1.5.2	lib.res.on.arguments	Restrictions on arguments
17.1.5.3	lib.res.on.exception.handling	Restrictions on exception handling
17.1.5.4	lib.alternate.definitions.for.functions	Alternate definitions for functions
17.1.5.5	lib.objects.within.classes	Objects within classes
17.1.5.6	lib.optional.members	Optional members
17.1.5.7	lib.functions.within.classes	Functions within classes
17.1.5.8	lib.global.functions	Global functions
17.1.5.9	lib.unreserved.names	Unreserved names
17.1.5.10	lib.implementation.types	Implementation types
17.1.5.10.1	lib.enumerated.types	Enumerated types
17.1.5.10.2	lib.bitmask.types	Bitmask types
17.1.5.10.3	lib.derived.classes	Derived classes

17.1.5.11	lib.protection.within.classes	Protection within classes
17.1.5.12	lib.definitions	Definitions
17.2	lib.standard.c.library	Standard C library
17.2.1	lib.mods.to.headers	Modifications to headers
17.2.2	lib.mods.to.definitions	Modifications to definitions
17.2.2.1	lib.wchar.t	Type `wchar_t`
17.2.2.2	lib.null	Macro `NULL`
17.2.2.3	lib.header.iso646.h	Header `<iso646.h>`
17.2.3	lib.mods.to.declarations	Modifications to declarations
17.2.3.1	lib.memchr	`memchr(const void*, int, size_t)`
17.2.3.2	lib.strchr	`strchr(const char*, int)`
17.2.3.3	lib.strpbrk	`strpbrk(const char*, const char*)`
17.2.3.4	lib.strrchr	`strrchr(const char*, int)`
17.2.3.5	lib.strstr	`strstr(const char*, const char*)`
17.2.4	lib.mods.to.behavior	Modifications to behavior
17.2.4.1	lib.offsetof	Macro `offsetof`
17.2.4.2	lib.longjmp	`longjmp(jmp_buf, int)`
17.2.4.3	lib.storage.allocation.functions	Storage allocation functions
17.2.4.4	lib.exit	`exit(int)`
17.3	lib.language.support	Language support
17.3.1	lib.header.defines	Header `<defines>`
17.3.1.1	lib.fvoid.t	Type `fvoid_t`
17.3.1.2	lib.ptrdiff.t	Type `ptrdiff_t`
17.3.1.3	lib.size.t	Type `size_t`
17.3.1.4	lib.wint.t	Type `wint_t`
17.3.1.5	lib.capacity	Type `capacity`
17.3.2	lib.header.exception	Header `<exception>`
17.3.2.1	lib.xmsg	Class `xmsg`
17.3.2.1.1	lib.xmsg::raise.handler	Type `xmsg::raise_handler`
17.3.2.1.2	lib.xmsg::set.raise.handler	`xmsg::set_raise_handler(raise_handler)`
17.3.2.1.3	lib.cons.xmsg.sss	`xmsg::xmsg(const char*, const char*, const char*)`
17.3.2.1.4	lib.des.xmsg	`xmsg::~xmsg()`
17.3.2.1.5	lib.xmsg::raise	`xmsg::raise()`
17.3.2.1.6	lib.xmsg::what	`xmsg::what()`
17.3.2.1.7	lib.xmsg::where	`xmsg::where()`
17.3.2.1.8	lib.xmsg::why	`xmsg::why()`
17.3.2.1.9	lib.xmsg::do.raise	`xmsg::do_raise()`
17.3.2.1.10	lib.cons.xmsg.sssi	`xmsg::xmsg(const char*, const char*, const char*, int)`
17.3.2.2	lib.xlogic	Class `xlogic`
17.3.2.2.1	lib.cons.xlogic	`xlogic::xlogic(const char*, const char*, const char*)`
17.3.2.2.2	lib.des.xlogic	`xlogic::~xlogic()`
17.3.2.2.3	lib.xlogic::do.raise	`xlogic::do_raise()`
17.3.2.3	lib.xruntime	Class `xruntime`
17.3.2.3.1	lib.cons.xruntime.sss	`xruntime::xruntime(const char*, const char*, const char*)`
17.3.2.3.2	lib.des.xruntime	`xruntime::~xruntime()`
17.3.2.3.3	lib.xruntime::do.raise	`xruntime::do_raise()`
17.3.2.3.4	lib.cons.xruntime.sssi	`xruntime::xruntime(const char*, const char*, const char*, int)`
17.3.2.4	lib.badcast	Class `badcast`
17.3.2.4.1	lib.cons.badcast	`badcast::badcast()`
17.3.2.4.2	lib.des.badcast	`badcast::~badcast()`

17.3.2.4.3	lib.badcast::do.raise	badcast::do_raise()
17.3.2.5	lib.invalidargument	Class invalidargument
17.3.2.5.1	lib.cons.invalidargument	invalidargument::invalidargument(const char*, const char*)
17.3.2.5.2	lib.des.invalidargument	invalidargument::~invalidargument()
17.3.2.5.3	lib.invalidargument::do.raise	invalidargument::do_raise()
17.3.2.6	lib.lengtherror	Class lengtherror
17.3.2.6.1	lib.cons.lengtherror	lengtherror::lengtherror(const char*, const char*)
17.3.2.6.2	lib.des.lengtherror	lengtherror::~lengtherror()
17.3.2.6.3	lib.lengtherror::do.raise	lengtherror::do_raise()
17.3.2.7	lib.outofrange	Class outofrange
17.3.2.7.1	lib.cons.outofrange	outofrange::outofrange(const char*, const char*)
17.3.2.7.2	lib.des.outofrange	outofrange::~outofrange()
17.3.2.7.3	lib.outofrange::do.raise	outofrange::do_raise()
17.3.2.8	lib.overflow	Class overflow
17.3.2.8.1	lib.cons.overflow	overflow::overflow(const char*, const char*)
17.3.2.8.2	lib.des.overflow	overflow::~overflow()
17.3.2.8.3	lib.overflow::do.raise	overflow::do_raise()
17.3.2.9	lib.xdomain	Class xdomain
17.3.2.9.1	lib.cons.xdomain	xdomain::xdomain(const char*, const char*, const char*)
17.3.2.9.2	lib.des.xdomain	xdomain::~xdomain()
17.3.2.9.3	lib.xdomain::do.raise	xdomain::do_raise()
17.3.2.10	lib.xrange	Class xrange
17.3.2.10.1	lib.cons.xrange	xrange::xrange(const char*, const char*, const char*)
17.3.2.10.2	lib.des.xrange	xrange::~xrange()
17.3.2.10.3	lib.xrange::do.raise	xrange::do_raise()
17.3.2.11	lib.set.terminate	set_terminate(fvoid_t*)
17.3.2.12	lib.set.unexpected	set_unexpected(fvoid_t*)
17.3.2.13	lib.terminate	terminate()
17.3.2.14	lib.unexpected	unexpected()
17.3.3	lib.header.new	Header <new>
17.3.3.1	lib.xalloc	Class xalloc
17.3.3.1.1	lib.cons.xalloc	xalloc::xalloc(const char*, const char*)
17.3.3.1.2	lib.des.xalloc	xalloc::~xalloc()
17.3.3.1.3	lib.xalloc::do.raise	xalloc::do_raise()
17.3.3.2	lib.set.new.handler	set_new_handler(fvoid_t*)
17.3.3.3	lib.op.delete	operator delete(void*)
17.3.3.4	lib.op.delete.array	operator delete[](void*)
17.3.3.5	lib.op.new	operator new(size_t)
17.3.3.6	lib.op.new.array	operator new[](size_t)
17.3.3.7	lib.placement.op.new	operator new(size_t, void*)
17.3.3.8	lib.placement.op.new.array	operator new[](size_t, void*)
17.3.4	lib.header.typeinfo	Header <typeinfo>
17.3.4.1	lib.badtypeid	Class badtypeid
17.3.4.1.1	lib.cons.badtypeid	badtypeid::badtypeid()
17.3.4.1.2	lib.des.badtypeid	badtypeid::~badtypeid()
17.3.4.1.3	lib.badtypeid::do.raise	badtypeid::do_raise()
17.3.4.2	lib.typeinfo	Class typeinfo
17.3.4.2.1	lib.des.typeinfo	typeinfo::~typeinfo()

17.3.4.2.2	lib.typeinfo::op==	`typeinfo::operator==(const typeinfo&)`
17.3.4.2.3	lib.typeinfo::op!=	`typeinfo::operator!=(const typeinfo&)`
17.3.4.2.4	lib.typeinfo::before	`typeinfo::before(const typeinfo&)`
17.3.4.2.5	lib.typeinfo::name	`typeinfo::name()`
17.3.4.2.6	lib.cons.typeinfo	`typeinfo::typeinfo(const typeinfo&)`
17.3.4.2.7	lib.typeinfo::op=	`typeinfo::operator=(const typeinfo&)`
17.4	lib.input/output	Input/output
17.4.1	lib.header.ios	Header `<ios>`
17.4.1.1	lib.ios	Class `ios`
17.4.1.1.1	lib.ios::failure	Class `ios::failure`
17.4.1.1.1.1	lib.cons.ios::failure	`ios::failure::failure(const char*, const char*)`
17.4.1.1.1.2	lib.des.ios::failure	`ios::failure::~failure()`
17.4.1.1.1.3	lib.ios::failure::do.raise	`ios::failure::do_raise()`
17.4.1.1.2	lib.ios::fmtflags	Type `ios::fmtflags`
17.4.1.1.3	lib.ios::iostate	Type `ios::iostate`
17.4.1.1.4	lib.ios::openmode	Type `ios::openmode`
17.4.1.1.5	lib.ios::seekdir	Type `ios::seekdir`
17.4.1.1.6	lib.ios::io.state	Type `ios::io_state`
17.4.1.1.7	lib.ios::open.mode	Type `ios::open_mode`
17.4.1.1.8	lib.ios::seek.dir	Type `ios::seek_dir`
17.4.1.1.9	lib.ios::init	Class `ios::Init`
17.4.1.1.9.1	lib.cons.ios::init	`ios::Init::Init()`
17.4.1.1.9.2	lib.des.ios::init	`ios::Init::~Init()`
17.4.1.1.10	lib.cons.ios.sb	`ios::ios(streambuf*)`
17.4.1.1.11	lib.des.ios	`ios::~ios()`
17.4.1.1.12	lib.ios::operator.void*	`ios::operator void*()`
17.4.1.1.13	lib.ios::operator!	`ios::operator!()`
17.4.1.1.14	lib.ios::copyfmt	`ios::copyfmt(const ios&)`
17.4.1.1.15	lib.ios::tie	`ios::tie()`
17.4.1.1.16	lib.ios::tie.os	`ios::tie(ostream*)`
17.4.1.1.17	lib.ios::rdbuf	`ios::rdbuf()`
17.4.1.1.18	lib.ios::rdbuf.sb	`ios::rdbuf(streambuf*)`
17.4.1.1.19	lib.ios::rdstate	`ios::rdstate()`
17.4.1.1.20	lib.ios::clear.ios	`ios::clear(iostate)`
17.4.1.1.21	lib.ios::clear.ios.old	`ios::clear(io_state)`
17.4.1.1.22	lib.ios::setstate.ios	`ios::setstate(iostate)`
17.4.1.1.23	lib.ios::setstate.ios.old	`ios::setstate(io_state)`
17.4.1.1.24	lib.ios::good	`ios::good()`
17.4.1.1.25	lib.ios::eof	`ios::eof()`
17.4.1.1.26	lib.ios::fail	`ios::fail()`
17.4.1.1.27	lib.ios::bad	`ios::bad()`
17.4.1.1.28	lib.ios::exceptions	`ios::exceptions()`
17.4.1.1.29	lib.ios::exceptions.ios	`ios::exceptions(iostate)`
17.4.1.1.30	lib.ios::exceptions.ios.old	`ios::exceptions(io_state)`
17.4.1.1.31	lib.ios::flags	`ios::flags()`
17.4.1.1.32	lib.ios::flags.f	`ios::flags(fmtflags)`
17.4.1.1.33	lib.ios::setf.f	`ios::setf(fmtflags)`
17.4.1.1.34	lib.ios::setf.ff	`ios::setf(fmtflags, fmtflags)`
17.4.1.1.35	lib.ios::unsetf	`ios::unsetf(fmtflags)`
17.4.1.1.36	lib.ios::fill	`ios::fill()`
17.4.1.1.37	lib.ios::fill.i	`ios::fill(int)`
17.4.1.1.38	lib.ios::precision	`ios::precision()`
17.4.1.1.39	lib.ios::precision.i	`ios::precision(int)`
17.4.1.1.40	lib.ios::width	`ios::width()`

17.4.1.1.41	lib.ios::width.i	`ios::width(int)`
17.4.1.1.42	lib.ios::xalloc	`ios::xalloc()`
17.4.1.1.43	lib.ios::iword	`ios::iword(int)`
17.4.1.1.44	lib.ios::pword	`ios::pword(int)`
17.4.1.1.45	lib.cons.ios	`ios::ios()`
17.4.1.1.46	lib.ios::init.sb	`ios::init(streambuf*)`
17.4.1.2	lib.dec	`dec(ios&)`
17.4.1.3	lib.fixed	`fixed(ios&)`
17.4.1.4	lib.hex	`hex(ios&)`
17.4.1.5	lib.internal	`internal(ios&)`
17.4.1.6	lib.left	`left(ios&)`
17.4.1.7	lib.noshowbase	`noshowbase(ios&)`
17.4.1.8	lib.noshowpoint	`noshowpoint(ios&)`
17.4.1.9	lib.noshowpos	`noshowpos(ios&)`
17.4.1.10	lib.noskipws	`noskipws(ios&)`
17.4.1.11	lib.nouppercase	`nouppercase(ios&)`
17.4.1.12	lib.oct	`oct(ios&)`
17.4.1.13	lib.right	`right(ios&)`
17.4.1.14	lib.scientific	`scientific(ios&)`
17.4.1.15	lib.showbase	`showbase(ios&)`
17.4.1.16	lib.showpoint	`showpoint(ios&)`
17.4.1.17	lib.showpos	`showpos(ios&)`
17.4.1.18	lib.skipws	`skipws(ios&)`
17.4.1.19	lib.uppercase	`uppercase(ios&)`
17.4.2	lib.header.streambuf	Header `<streambuf>`
17.4.2.1	lib.streamoff	Type `streamoff`
17.4.2.2	lib.streampos	Class `streampos`
17.4.2.2.1	lib.cons.streampos	`streampos::streampos(streamoff)`
17.4.2.2.2	lib.streampos::offset	`streampos::offset()`
17.4.2.2.3	lib.streampos::op-.sp	`streampos::operator-(streampos&)`
17.4.2.2.4	lib.streampos::op+=	`streampos::operator+=(streamoff)`
17.4.2.2.5	lib.streamos::op-=	`streampos::operator-=(streamoff)`
17.4.2.2.6	lib.streampos::op+	`streampos::operator+(streamoff)`
17.4.2.2.7	lib.streampos::op-.off	`streampos::operator-(streamoff)`
17.4.2.2.8	lib.streampos::op==	`streampos::operator==(streampos&)`
17.4.2.2.9	lib.op!=.streampos	`operator!=(streampos&)`
17.4.2.3	lib.streambuf	Class `streambuf`
17.4.2.3.1	lib.des.streambuf	`streambuf::~streambuf()`
17.4.2.3.2	lib.streambuf::pubseekoff	`streambuf::pubseekoff(streamoff, ios::seekdir, ios::openmode)`
17.4.2.3.3	lib.streambuf::pubseekoff.old	`streambuf::pubseekoff(streamoff, ios::seek_dir, ios::open_mode)`
17.4.2.3.4	lib.streambuf::pubseekpos	`streambuf::pubseekpos(streampos, ios::openmode)`
17.4.2.3.5	lib.streambuf::pubseekpos.old	`streambuf::pubseekpos(streampos, ios::open_mode)`
17.4.2.3.6	lib.streambuf::pubsetbuf	`streambuf::pubsetbuf(char*, int)`
17.4.2.3.7	lib.streambuf::pubsync	`streambuf::pubsync()`
17.4.2.3.8	lib.streambuf::sbumpc	`streambuf::sbumpc()`
17.4.2.3.9	lib.streambuf::sgetc	`streambuf::sgetc()`
17.4.2.3.10	lib.streambuf::sgetn	`streambuf::sgetn(char*, int)`
17.4.2.3.11	lib.streambuf::snextc	`streambuf::snextc()`
17.4.2.3.12	lib.streambuf::sputbackc	`streambuf::sputbackc(char)`
17.4.2.3.13	lib.streambuf::sungetc	`streambuf::sungetc()`
17.4.2.3.14	lib.streambuf::sputc	`streambuf::sputc(int)`

17.4.2.3.15	lib.streambuf::sputn	streambuf::sputn(const char*, int)
17.4.2.3.16	lib.cons.streambuf	streambuf::streambuf()
17.4.2.3.17	lib.streambuf::eback	streambuf::eback()
17.4.2.3.18	lib.streambuf::gptr	streambuf::gptr()
17.4.2.3.19	lib.streambuf::egptr	streambuf::egptr()
17.4.2.3.20	lib.streambuf::gbump	streambuf::gbump(int)
17.4.2.3.21	lib.streambuf::setg	streambuf::setg(char*, char*, char*)
17.4.2.3.22	lib.streambuf::pbase	streambuf::pbase()
17.4.2.3.23	lib.streambuf::pptr	streambuf::pptr()
17.4.2.3.24	lib.streambuf::epptr	streambuf::epptr()
17.4.2.3.25	lib.streambuf::pbump	streambuf::pbump(int)
17.4.2.3.26	lib.streambuf::setp	streambuf::setp(char*, char*)
17.4.2.3.27	lib.streambuf::overflow	streambuf::overflow(int)
17.4.2.3.28	lib.streambuf::pbackfail	streambuf::pbackfail(int)
17.4.2.3.29	lib.streambuf::underflow	streambuf::underflow()
17.4.2.3.30	lib.streambuf::uflow	streambuf::uflow()
17.4.2.3.31	lib.streambuf::xsgetn	streambuf::xsgetn(char*, int)
17.4.2.3.32	lib.streambuf::xsputn	streambuf::xsputn(const char*, int)
17.4.2.3.33	lib.streambuf::seekoff	streambuf::seekoff(streamoff, ios::seekdir, ios::openmode)
17.4.2.3.34	lib.streambuf::seekpos	streambuf::seekpos(streampos, ios::openmode)
17.4.2.3.35	lib.streambuf::setbuf	streambuf::setbuf(char*, int)
17.4.2.3.36	lib.streambuf::sync	streambuf::sync()
17.4.3	lib.header.istream	Header <istream>
17.4.3.1	lib.istream	Class istream
17.4.3.1.1	lib.cons.istream	istream::istream()
17.4.3.1.2	lib.des.istream	istream::~istream()
17.4.3.1.3	lib.istream::ipfx	istream::ipfx(int)
17.4.3.1.4	lib.istream::isfx	istream::isfx()
17.4.3.1.5	lib.istream::ext.imanip	istream::operator>>(istream& (*)(istream&))
17.4.3.1.6	lib.istream::ext.iomanip	istream::operator>>(ios& (*)(ios&))
17.4.3.1.7	lib.istream::ext.str	istream::operator>>(char*)
17.4.3.1.8	lib.istream::ext.ustr	istream::operator>>(unsigned char*)
17.4.3.1.9	lib.istream::ext.sstr	istream::operator>>(signed char*)
17.4.3.1.10	lib.istream::ext.c	istream::operator>>(char&)
17.4.3.1.11	lib.istream::ext.uc	istream::operator>>(unsigned char&)
17.4.3.1.12	lib.istream::ext.sc	istream::operator>>(signed char&)
17.4.3.1.13	lib.istream::ext.si	istream::operator>>(short&)
17.4.3.1.14	lib.istream::ext.usi	istream::operator>>(unsigned short&)
17.4.3.1.15	lib.istream::ext.i	istream::operator>>(int&)
17.4.3.1.16	lib.istream::ext.ui	istream::operator>>(unsigned int&)
17.4.3.1.17	lib.istream::ext.li	istream::operator>>(long&)
17.4.3.1.18	lib.istream::ext.uli	istream::operator>>(unsigned long&)
17.4.3.1.19	lib.istream::ext.f	istream::operator>>(float&)
17.4.3.1.20	lib.istream::ext.d	istream::operator>>(double&)
17.4.3.1.21	lib.istream::ext.ld	istream::operator>>(long double&)
17.4.3.1.22	lib.istream::ext.ptr	istream::operator>>(void*&)
17.4.3.1.23	lib.istream::ext.sb	istream::operator>>(streambuf&)
17.4.3.1.24	lib.istream::get	istream::get()
17.4.3.1.25	lib.istream::get.str	istream::get(char*, int, char)
17.4.3.1.26	lib.istream::get.ustr	istream::get(unsigned char*, int, char)
17.4.3.1.27	lib.istream::get.sstr	istream::get(signed char*, int,

		char)
17.4.3.1.28	lib.istream::get.c	istream::get(char&)
17.4.3.1.29	lib.istream::get.uc	istream::get(unsigned char&)
17.4.3.1.30	lib.istream::get.sc	istream::get(signed char&)
17.4.3.1.31	lib.istream::get.sb	istream::get(streambuf&, char)
17.4.3.1.32	lib.istream::getline.str	istream::getline(char*, int, char)
17.4.3.1.33	lib.istream::getline.ustr	istream::getline(unsigned char*, int, char)
17.4.3.1.34	lib.istream::getline.sstr	istream::getline(signed char*, int, char)
17.4.3.1.35	lib.istream::ignore	istream::ignore(int, int)
17.4.3.1.36	lib.istream::read.str	istream::read(char*, int)
17.4.3.1.37	lib.istream::read.ustr	istream::read(unsigned char*, int)
17.4.3.1.38	lib.istream::read.sstr	istream::read(signed char*, int)
17.4.3.1.39	lib.istream::peek	istream::peek()
17.4.3.1.40	lib.istream::putback	istream::putback(char)
17.4.3.1.41	lib.istream::unget	istream::unget()
17.4.3.1.42	lib.istream::gcount	istream::gcount()
17.4.3.1.43	lib.istream::sync	istream::sync()
17.4.3.2	lib.ws	ws(istream&)
17.4.4	lib.header.ostream	Header <ostream>
17.4.4.1	lib.ostream	Class ostream
17.4.4.1.1	lib.cons.ostream.sb	ostream::ostream(streambuf*)
17.4.4.1.2	lib.des.ostream	ostream::~ostream()
17.4.4.1.3	lib.ostream::opfx	ostream::opfx()
17.4.4.1.4	lib.ostream::osfx	ostream::osfx()
17.4.4.1.5	lib.ostream::ins.omanip	ostream::operator<<(ostream& (*)(ostream&))
17.4.4.1.6	lib.ostream::ins.iomanip	ostream::operator<<(ios& (*)(ios&))
17.4.4.1.7	lib.ostream::ins.str	ostream::operator<<(const char*)
17.4.4.1.8	lib.ostream::ins.c	ostream::operator<<(char)
17.4.4.1.9	lib.ostream::ins.uc	ostream::operator<<(unsigned char)
17.4.4.1.10	lib.ostream::ins.sc	ostream::operator<<(signed char)
17.4.4.1.11	lib.ostream::ins.si	ostream::operator<<(short)
17.4.4.1.12	lib.ostream::ins.usi	ostream::operator<<(unsigned short)
17.4.4.1.13	lib.ostream::ins.i	ostream::operator<<(int)
17.4.4.1.14	lib.ostream::ins.ui	ostream::operator<<(unsigned int)
17.4.4.1.15	lib.ostream::ins.li	ostream::operator<<(long)
17.4.4.1.16	lib.ostream::ins.uli	ostream::operator<<(unsigned long)
17.4.4.1.17	lib.ostream::ins.f	ostream::operator<<(float)
17.4.4.1.18	lib.ostream::ins.d	ostream::operator<<(double)
17.4.4.1.19	lib.ostream::ins.ld	ostream::operator<<(long double)
17.4.4.1.20	lib.ostream::ins.ptr	ostream::operator<<(void*)
17.4.4.1.21	lib.ostream::ins.sb	ostream::operator<<(streambuf&)
17.4.4.1.22	lib.ostream::put	ostream::put(char)
17.4.4.1.23	lib.ostream::write.str	ostream::write(const char*, int)
17.4.4.1.24	lib.ostream::write.ustr	ostream::write(const unsigned char*, int)
17.4.4.1.25	lib.ostream::write.sstr	ostream::write(const signed char*, int)
17.4.4.1.26	lib.ostream::flush	ostream::flush()
17.4.4.2	lib.endl	endl(ostream&)
17.4.4.3	lib.ends	ends(ostream&)
17.4.4.4	lib.flush	flush(ostream&)
17.4.5	lib.header.iomanip	Header <iomanip>

17.4.5.1	lib.template.smanip	Template class smanip<T>
17.4.5.1.1	lib.cons.smanip.ios	smanip<T>::smanip(ios& (*)(ios&, T), T
17.4.5.1.2	lib.ext.smanip	operator>>(istream&, const smanip<T>&)
17.4.5.1.3	lib.ins.smanip	operator<<(ostream&, const smanip<T>&)
17.4.5.2	lib.template.imanip	Template class imanip<T>
17.4.5.2.1	lib.cons.imanip.ios	imanip<T>::imanip(ios& (*)(ios&, T), T
17.4.5.2.2	lib.ext.imanip	operator>>(istream&, const imanip<T>&)
17.4.5.3	lib.template.omanip	Template class omanip<T>
17.4.5.3.1	lib.cons.omanip.ios	omanip<T>::omanip(ios& (*)(ios&, T), T
17.4.5.3.2	lib.ins.omanip	operator<<(istream&, const omanip<T>&)
17.4.5.4	lib.instantiations.of.manipulators	Instantiations of manipulators
17.4.5.4.1	lib.resetiosflags	resetiosflags(ios::fmtflags)
17.4.5.4.2	lib.setiosflags	setiosflags(ios::fmtflags)
17.4.5.4.3	lib.setbase	setbase(int)
17.4.5.4.4	lib.setfill	setfill(int)
17.4.5.4.5	lib.setprecision	setprecision(int)
17.4.5.4.6	lib.setw	setw(int)
17.4.6	lib.header.strstream	Header <strstream>
17.4.6.1	lib.strstreambuf	Class strstreambuf
17.4.6.1.1	lib.cons.strstreambuf.i	strstreambuf::strstreambuf(int)
17.4.6.1.2	lib.cons.strstreambuf.ff	strstreambuf::strstreambuf(void* (*)(size_t), void (*)(void*))
17.4.6.1.3	lib.cons.strstreambuf.str	strstreambuf::strstreambuf(char*, int, char*)
17.4.6.1.4	lib.cons.strstreambuf.ustr	strstreambuf::strstreambuf(unsigned char*, int, unsigned char*)
17.4.6.1.5	lib.cons.strstreambuf.sstr	strstreambuf::strstreambuf(signed char*, int, signed char*)
17.4.6.1.6	lib.cons.strstreambuf.cstr	strstreambuf::strstreambuf(const char*, int)
17.4.6.1.7	lib.cons.strstreambuf.custr	strstreambuf::strstreambuf(const unsigned char*, int)
17.4.6.1.8	lib.cons.strstreambuf.csstr	strstreambuf::strstreambuf(const signed char*, int)
17.4.6.1.9	lib.des.strstreambuf	strstreambuf::~strstreambuf()
17.4.6.1.10	lib.strstreambuf::freeze	strstreambuf::freeze(int)
17.4.6.1.11	lib.strstreambuf::str	strstreambuf::str()
17.4.6.1.12	lib.strstreambuf::pcount	strstreambuf::pcount()
17.4.6.1.13	lib.strstreambuf::overflow	strstreambuf::overflow(int)
17.4.6.1.14	lib.strstreambuf::pbackfail	strstreambuf::pbackfail(int)
17.4.6.1.15	lib.strstreambuf::underflow	strstreambuf::underflow()
17.4.6.1.16	lib.strstreambuf::uflow	strstreambuf::uflow()
17.4.6.1.17	lib.strstreambuf::xsgetn	strstreambuf::xsgetn(char*, int)
17.4.6.1.18	lib.strstreambuf::xsputn	strstreambuf::xsputn(const char*, int)
17.4.6.1.19	lib.strstreambuf::seekoff	strstreambuf::seekoff(streamoff, ios::seekdir, ios::openmode)

17.4.6.1.20	lib.strstreambuf::seekpos	strstreambuf::seekpos(streampos, ios::openmode)
17.4.6.1.21	lib.strstreambuf::setbuf	strstreambuf::setbuf(char*, int)
17.4.6.1.22	lib.strstreambuf::sync	strstreambuf::sync()
17.4.6.2	lib.istrstream	Class istrstream
17.4.6.2.1	lib.cons.istrstream.cstr	istrstream::istrstream(const char*)
17.4.6.2.2	lib.cons.istrstream.cstrn	istrstream::istrstream(const char*, int)
17.4.6.2.3	lib.cons.istrstream.str	istrstream::istrstream(char*)
17.4.6.2.4	lib.cons.istrstream.strn	istrstream::istrstream(char*, int)
17.4.6.2.5	lib.des.istrstream	istrstream::~istrstream()
17.4.6.2.6	lib.istrstream::rdbuf	istrstream::rdbuf()
17.4.6.3	lib.ostrstream	Class ostrstream
17.4.6.3.1	lib.cons.ostrstream	ostrstream::ostrstream()
17.4.6.3.2	lib.cons.ostrstream.str	ostrstream::ostrstream(char*, int, openmode)
17.4.6.3.3	lib.des.ostrstream	ostrstream::~ostrstream()
17.4.6.3.4	lib.ostrstream::rdbuf	ostrstream::rdbuf()
17.4.6.3.5	lib.ostrstream::freeze	ostrstream::freeze(int)
17.4.6.3.6	lib.ostrstream::str	ostrstream::str()
17.4.6.3.7	lib.ostrstream::pcount	ostrstream::pcount()
17.4.7	lib.header.sstream	Header <sstream>
17.4.7.1	lib.stringbuf	Class stringbuf
17.4.7.1.1	lib.cons.stringbuf.m	stringbuf::stringbuf(ios::openmode)
17.4.7.1.2	lib.cons.stringbuf.sm	stringbuf::stringbuf(const string&, ios::openmode)
17.4.7.1.3	lib.des.stringbuf	stringbuf::~stringbuf()
17.4.7.1.4	lib.stringbuf::str	stringbuf::str()
17.4.7.1.5	lib.stringbuf::str.s	stringbuf::str(const string&)
17.4.7.1.6	lib.stringbuf::overflow	stringbuf::overflow(int)
17.4.7.1.7	lib.stringbuf::pbackfail	stringbuf::pbackfail(int)
17.4.7.1.8	lib.stringbuf::underflow	stringbuf::underflow()
17.4.7.1.9	lib.stringbuf::uflow	stringbuf::uflow()
17.4.7.1.10	lib.stringbuf::xsgetn	stringbuf::xsgetn(char*, int)
17.4.7.1.11	lib.stringbuf::xsputn	stringbuf::xsputn(const char*, int)
17.4.7.1.12	lib.stringbuf::seekoff	stringbuf::seekoff(streamoff, ios::seekdir, ios::openmode)
17.4.7.1.13	lib.stringbuf::seekpos	stringbuf::seekpos(streampos, ios::openmode)
17.4.7.1.14	lib.stringbuf::setbuf	stringbuf::setbuf(char*, int)
17.4.7.1.15	lib.stringbuf::sync	stringbuf::sync()
17.4.7.2	lib.istringstream	Class istringstream
17.4.7.2.1	lib.cons.istringstream.m	istringstream::istringstream(ios::openmode)
17.4.7.2.2	lib.cons.istringstream.sm	istringstream::istringstream(const string&, ios::openmode
17.4.7.2.3	lib.des.istringstream	istringstream::~istringstream()
17.4.7.2.4	lib.istringstream::rdbuf	istringstream::rdbuf()
17.4.7.2.5	lib.istringstream::str	istringstream::str()
17.4.7.2.6	lib.istringstream::str.s	istringstream::str(const string&)
17.4.7.3	lib.ostringstream	Class ostringstream
17.4.7.3.1	lib.cons.ostringstream.m	ostringstream::ostringstream(ios::openmode)
17.4.7.3.2	lib.cons.ostringstream.sm	ostringstream::ostringstream(const string&, ios::openmode
17.4.7.3.3	lib.des.ostringstream	ostringstream::~ostringstream()
17.4.7.3.4	lib.ostringstream::rdbuf	ostringstream::rdbuf()

17.4.7.3.5	lib.ostringstream::str	ostringstream::str()
17.4.7.3.6	lib.ostringstream::str.s	ostringstream::str(const string&)
17.4.8	lib.header.fstream	Header <fstream>
17.4.8.1	lib.filebuf	Class filebuf
17.4.8.1.1	lib.cons.filebuf	filebuf::filebuf()
17.4.8.1.2	lib.des.filebuf	filebuf::~filebuf()
17.4.8.1.3	lib.filebuf::is.open	filebuf::is_open()
17.4.8.1.4	lib.filebuf::open	filebuf::open(const char*, ios::openmode)
17.4.8.1.5	lib.filebuf::open.old	filebuf::open(const char*, ios::open_mode)
17.4.8.1.6	lib.filebuf::close	filebuf::close()
17.4.8.1.7	lib.filebuf::overflow	filebuf::overflow(int)
17.4.8.1.8	lib.filebuf::pbackfail	filebuf::pbackfail(int)
17.4.8.1.9	lib.filebuf::underflow	filebuf::underflow()
17.4.8.1.10	lib.filebuf::uflow	filebuf::uflow()
17.4.8.1.11	lib.filebuf::xsgetn	filebuf::xsgetn(char*, int)
17.4.8.1.12	lib.filebuf::xsputn	filebuf::xsputn(const char*, int)
17.4.8.1.13	lib.filebuf::seekoff	filebuf::seekoff(streamoff, ios::seekdir, ios::openmode)
17.4.8.1.14	lib.filebuf::seekpos	filebuf::seekpos(streampos, ios::openmode)
17.4.8.1.15	lib.filebuf::setbuf	filebuf::setbuf(char*, int)
17.4.8.1.16	lib.filebuf::sync	filebuf::sync()
17.4.8.2	lib.ifstream	Class ifstream
17.4.8.2.1	lib.cons.ifstream	ifstream::ifstream()
17.4.8.2.2	lib.cons.ifstream.fn	ifstream::ifstream(const char*, openmode)
17.4.8.2.3	lib.des.ifstream	ifstream::~ifstream()
17.4.8.2.4	lib.ifstream::rdbuf	ifstream::rdbuf()
17.4.8.2.5	lib.ifstream::is.open	ifstream::is_open()
17.4.8.2.6	lib.ifstream::open	ifstream::open(const char*, openmode)
17.4.8.2.7	lib.ifstream::open.old	ifstream::open(const char*, open_mode)
17.4.8.2.8	lib.ifstream::close	ifstream::close()
17.4.8.3	lib.ofstream	Class ofstream
17.4.8.3.1	lib.cons.ofstream	ofstream::ofstream()
17.4.8.3.2	lib.cons.ofstream.fn	ofstream::ofstream(const char*, openmode)
17.4.8.3.3	lib.des.ofstream	ofstream::~ofstream()
17.4.8.3.4	lib.ofstream::rdbuf	ofstream::rdbuf()
17.4.8.3.5	lib.ofstream::is.open	ofstream::is_open()
17.4.8.3.6	lib.ofstream::open	ofstream::open(const char*, openmode)
17.4.8.3.7	lib.ofstream::open.old	ofstream::open(const char*, open_mode)
17.4.8.3.8	lib.ofstream::close	ofstream::close()
17.4.8.4	lib.stdiobuf	Class stdiobuf
17.4.8.4.1	lib.cons.stdiobuf.fi	stdiobuf::stdiobuf(*FILE**)
17.4.8.4.2	lib.des.stdiobuf	stdiobuf::~stdiobuf()
17.4.8.4.3	lib.stdiobuf::buffered	stdiobuf::buffered()
17.4.8.4.4	lib.stdiobuf::buffered.i	stdiobuf::buffered(int)
17.4.8.4.5	lib.stdiobuf::overflow	stdiobuf::overflow(int)
17.4.8.4.6	lib.stdiobuf::pbackfail	stdiobuf::pbackfail(int)

17.4.8.4.7	lib.stdiobuf::underflow	stdiobuf::underflow()
17.4.8.4.8	lib.stdiobuf::uflow	stdiobuf::uflow()
17.4.8.4.9	lib.stdiobuf::xsgetn	stdiobuf::xsgetn(char*, int)
17.4.8.4.10	lib.stdiobuf::xsputn	stdiobuf::xsputn(const char*, int)
17.4.8.4.11	lib.stdiobuf::seekoff	stdiobuf::seekoff(streamoff, ios::seekdir, ios::openmode)
17.4.8.4.12	lib.stdiobuf::seekpos	stdiobuf::seekpos(streampos, ios::openmode)
17.4.8.4.13	lib.stdiobuf::setbuf	stdiobuf::setbuf(char*, int)
17.4.8.4.14	lib.stdiobuf::sync	stdiobuf::sync()
17.4.8.5	lib.istdiostream	Class istdiostream
17.4.8.5.1	lib.cons.istdiostream.fi	istdiostream::istdiostream(FILE*)
17.4.8.5.2	lib.des.istdiostream	istdiostream::~istdiostream()
17.4.8.5.3	lib.istdiostream::rdbuf	istdiostream::rdbuf()
17.4.8.5.4	lib.istdiostream::buffered	istdiostream::buffered()
17.4.8.5.5	lib.istdiostream::buffered.i	istdiostream::buffered(int)
17.4.8.6	lib.ostdiostream	Class ostdiostream
17.4.8.6.1	lib.cons.ostdiostream.fi	ostdiostream::ostdiostream(FILE*)
17.4.8.6.2	lib.des.ostdiostream	ostdiostream::~ostdiostream()
17.4.8.6.3	lib.ostdiostream::rdbuf	ostdiostream::rdbuf()
17.4.8.6.4	lib.ostdiostream::buffered	ostdiostream::buffered()
17.4.8.6.5	lib.ostdiostream::buffered.i	ostdiostream::buffered(int)
17.4.9	lib.header.iostream	Header <iostream>
17.4.9.1	lib.cin	Object cin
17.4.9.2	lib.cout	Object cout
17.4.9.3	lib.cerr	Object cerr
17.4.9.4	lib.clog	Object clog
17.5	lib.support.classes	Support classes
17.5.1	lib.header.string	Header <string>
17.5.1.1	lib.string	Class string
17.5.1.1.1	lib.cons.string	string::string()
17.5.1.1.2	lib.cons.string.cap	string::string(size_t, capacity)
17.5.1.1.3	lib.cons.string.sub	string::string(const string&, size_t, size_t)
17.5.1.1.4	lib.cons.string.str	string::string(const char*, size_t)
17.5.1.1.5	lib.cons.string.c	string::string(char, size_t)
17.5.1.1.6	lib.cons.string.uc	string::string(unsigned char, size_t)
17.5.1.1.7	lib.cons.string.sc	string(signed char, size_t)
17.5.1.1.8	lib.string::op=.str	string::operator=(const char*)
17.5.1.1.9	lib.string::op=.c	string::operator=(char)
17.5.1.1.10	lib.string::op+=.sub	string::operator+=(const string&)
17.5.1.1.11	lib.string::op+=.str	string::operator+=(const char*)
17.5.1.1.12	lib.string::op+=.c	string::operator+=(char)
17.5.1.1.13	lib.string::append.sub	string::append(const string&, size_t, size_t)
17.5.1.1.14	lib.string::append.str	string::append(const char*, size_t)
17.5.1.1.15	lib.string::append.c	string::append(char, size_t)
17.5.1.1.16	lib.string::assign.sub	string::assign(const string&, size_t, size_t)
17.5.1.1.17	lib.string::assign.str	string::assign(const char*, size_t)
17.5.1.1.18	lib.string::assign.c	string::assign(char, size_t)
17.5.1.1.19	lib.string::insert.sub	string::insert(size_t, const string&, size_t, size_t)
17.5.1.1.20	lib.string::insert.str	string::insert(size_t, const char*,

		size_t)
17.5.1.1.21	lib.string::insert.c	string::insert(size_t, char, size_t)
17.5.1.1.22	lib.string::remove	string::remove(size_t, size_t)
17.5.1.1.23	lib.string::replace.sub	string::replace(size_t, size_t, const string&, size_t, size_t)
17.5.1.1.24	lib.string::replace.str	string::replace(size_t, size_t, const char*, size_t)
17.5.1.1.25	lib.string::replace.c	string::replace(size_t, size_t, char, size_t)
17.5.1.1.26	lib.string::get.at	string::get_at(size_t)
17.5.1.1.27	lib.string::put.at	string::put_at(size_t, char)
17.5.1.1.28	lib.string::op.array	string::operator[](size_t)
17.5.1.1.29	lib.string::c.str	string::c_str()
17.5.1.1.30	lib.string::length	string::length()
17.5.1.1.31	lib.string::resize	string::resize(size_t, char)
17.5.1.1.32	lib.string::reserve	string::reserve()
17.5.1.1.33	lib.string::reserve.cap	string::reserve(size_t)
17.5.1.1.34	lib.string::copy	string::copy(char*, size_t, size_t)
17.5.1.1.35	lib.string::find.sub	string::find(const string&, size_t)
17.5.1.1.36	lib.string::find.str	string::find(const char*, size_t, size_t)
17.5.1.1.37	lib.string::find.c	string::find(char, size_t)
17.5.1.1.38	lib.string::rfind.sub	string::rfind(const string&, size_t)
17.5.1.1.39	lib.string::rfind.str	string::rfind(const char*, size_t, size_t)
17.5.1.1.40	lib.string::rfind.c	string::rfind(char, size_t)
17.5.1.1.41	lib.string::find.first.of.sub	string::find_first_of(const string&, size_t)
17.5.1.1.42	lib.string::find.first.of.str	string::find_first_of(const char*, size_t,size_t)
17.5.1.1.43	lib.string::find.first.of.c	string::find_first_of(char, size_t)
17.5.1.1.44	lib.string::find.last.of.sub	string::find_last_of(const string&, size_t)
17.5.1.1.45	lib.string::find.last.of.str	string::find_last_of(const char*, size_t,size_t)
17.5.1.1.46	lib.string::find.last.of.c	string::find_last_of(char, size_t)
17.5.1.1.47	lib.string::find.first.not.of.sub	string::find_first_not_of(const string&,size_t)
17.5.1.1.48	lib.string::find.first.not.of.str	string::find_first_not_of(const char*, size_t,size_t)
17.5.1.1.49	lib.string::find.first.not.of.c	string::find_first_not_of(char, size_t)
17.5.1.1.50	lib.string::find.last.not.of.sub	string::find_last_not_of(const string&, size_t)
17.5.1.1.51	lib.string::find.last.not.of.str	string::find_last_not_of(const char*, size_t, size_t)
17.5.1.1.52	lib.string::find.last.not.of.c	string::find_last_not_of(char, size_t)
17.5.1.1.53	lib.string::substr	string::substr(size_t, size_t)
17.5.1.1.54	lib.string::compare.sub	string::compare(const string&, size_t, size_t)
17.5.1.1.55	lib.string::compare.str	string::compare(const char*, size_t)
17.5.1.1.56	lib.string::compare.c	string::compare(char, size_t)
17.5.1.2	lib.op+.sub.sub	operator+(const string&, const string&)
17.5.1.3	lib.op+.str.sub	operator+(const char*,

		const string&)
17.5.1.4	lib.op+.c.sub	operator+(char, const string&)
17.5.1.5	lib.op+.sub.str	operator+(const string&, const char*)
17.5.1.6	lib.op+.str.c	operator+(const string&, char)
17.5.1.7	lib.op==.sub.sub	operator==(const string&, const string&)
17.5.1.8	lib.op==.str.sub	operator==(const char*, const string&)
17.5.1.9	lib.op==.c.sub	operator==(char, const string&)
17.5.1.10	lib.op==.sub.str	operator==(const string&, const char*)
17.5.1.11	lib.op==.sub.c	operator==(const string&, char)
17.5.1.12	lib.op!=.sub.sub	operator!=(const string&, const string&)
17.5.1.13	lib.op!=.str.sub	operator!=(const char*, const string&)
17.5.1.14	lib.op!=.c.sub	operator!=(char, const string&)
17.5.1.15	lib.op!=.sub.str	operator!=(const string&, const char*)
17.5.1.16	lib.op!=.sub.c	operator!=(const string&, char)
17.5.1.17	lib.ext.sub	operator>>(istream&, string&)
17.5.1.18	lib.getline.sub	getline(istream&, string&, char)
17.5.1.19	lib.ins.sub	operator<<(ostream&, const string&)
17.5.2	lib.header.wstring	Header <wstring>
17.5.2.1	lib.wstring	Class wstring
17.5.2.1.1	lib.cons.wstring	wstring::wstring()
17.5.2.1.2	lib.cons.wstring.cap	wstring::wstring(size_t, capacity)
17.5.2.1.3	lib.cons.wstring.wsub	wstring::wstring(const wstring&, size_t, size_t)
17.5.2.1.4	lib..cons.wstring.wstr	wstring::wstring(const wchar_t*, size_t)
17.5.2.1.5	lib..cons.wstring.wc	wstring::wstring(wchar_t, size_t)
17.5.2.1.6	lib.wstring::op=.wstr	wstring::operator=(const wchar_t*)
17.5.2.1.7	lib.wstring::op=.wc	wstring::operator=(wchar_t)
17.5.2.1.8	lib.wstring::op+=.wsub	wstring::operator+=(const wstring&)
17.5.2.1.9	lib.wstring::op+=.wstr	wstring::operator+=(const wchar_t*)
17.5.2.1.10	lib.wstring::op+=.wc	wstring::operator+=(wchar_t)
17.5.2.1.11	lib.wstring::append.wsub	wstring::append(const wstring&, size_t, size_t)
17.5.2.1.12	lib.wstring::append.wstr	wstring::append(const wchar_t*, size_t)
17.5.2.1.13	lib.wstring::append.wc	wstring::append(wchar_t, size_t)
17.5.2.1.14	lib.wstring::assign.wsub	wstring::assign(const wstring&, size_t, size_t)
17.5.2.1.15	lib.wstring::assign.wstr	wstring::assign(const wchar_t*, size_t)
17.5.2.1.16	lib.wstring::assign.wc	wstring::assign(wchar_t, size_t)
17.5.2.1.17	lib.wstring::insert.wsub	wstring::insert(size_t, const wstring&, size_t, size_t)
17.5.2.1.18	lib.wstring::insert.wstr	wstring::insert(size_t, const wchar_t*, size_t)
17.5.2.1.19	lib.wstring::insert.wc	wstring::insert(size_t, wchar_t, size_t)
17.5.2.1.20	lib.wstring::remove	wstring::remove(size_t, size_t)

17.5.2.1.21	lib.wstring::replace.wsub	`wstring::replace(size_t, size_t, const wstring&, size_t, size_t)`
17.5.2.1.22	lib.wstring::replace.wstr	`wstring::replace(size_t, size_t, const wchar_t*, size_t)`
17.5.2.1.23	lib.wstring::replace.wc	`wstring::replace(size_t, size_t, wchar_t, size_t)`
17.5.2.1.24	lib.wstring::get.at	`wstring::get_at(size_t)`
17.5.2.1.25	lib.wstring::put.at	`wstring::put_at(size_t, wchar_t)`
17.5.2.1.26	lib.wstring::op.array	`wstring::operator[](size_t)`
17.5.2.1.27	lib.wstring::c.wcs	`wstring::c_wcs()`
17.5.2.1.28	lib.wstring::length	`wstring::length()`
17.5.2.1.29	lib.wstring::resize	`wstring::resize(size_t, wchar_t)`
17.5.2.1.30	lib.wstring::reserve	`wstring::reserve()`
17.5.2.1.31	lib.wstring::reserve.cap	`wstring::reserve(size_t)`
17.5.2.1.32	lib.wstring::copy.wstr	`wstring::copy(wchar_t*, size_t, size_t)`
17.5.2.1.33	lib.wstring::find.wsub	`wstring::find(const wstring&, size_t)`
17.5.2.1.34	lib.wstring::find.wstr	`wstring::find(const wchar_t*, size_t, size_t)`
17.5.2.1.35	lib.wstring::find.wc	`wstring::find(wchar_t, size_t)`
17.5.2.1.36	lib.wstring::rfind.wsub	`wstring::rfind(const wstring&, size_t)`
17.5.2.1.37	lib.wstring::rfind.wstr	`wstring::rfind(const wchar_t*, size_t, size_t)`
17.5.2.1.38	lib.wstring::rfind.wc	`wstring::rfind(wchar_t, size_t)`
17.5.2.1.39	lib.wstring::find.first.of.wsub	`wstring::find_first_of(const wstring&, size_t)`
17.5.2.1.40	lib.wstring::find.first.of.wstr	`wstring::find_first_of(const wchar_t*, size_t, size_t)`
17.5.2.1.41	lib.wstring::find.first.of.wc	`wstring::find_first_of(wchar_t, size_t)`
17.5.2.1.42	lib.wstring::find.last.of.wsub	`wstring::find_last_of(const wstring&, size_t)`
17.5.2.1.43	lib.wstring::find.last.of.wstr	`wstring::find_last_of(const wchar_t*, size_t, size_t)`
17.5.2.1.44	lib.wstring::find.last.of.wc	`wstring::find_last_of(wchar_t, size_t)`
17.5.2.1.45	lib.wstring::find.first.not.of.wsub	`wstring::find_first_not_of(const wstring&, size_t)`
17.5.2.1.46	lib.wstring::find.first.not.of.wstr	`wstring::find_first_not_of(const wchar_t*, size_t, size_t)`
17.5.2.1.47	lib.wstring::find.first.not.of.wc	`wstring::find_first_not_of(wchar_t, size_t)`
17.5.2.1.48	lib.wstring::find.last.not.of.wsub	`wstring::find_last_not_of(const wstring&, size_t)`
17.5.2.1.49	lib.wstring::find.last.not.of.wstr	`wstring::find_last_not_of(const wchar_t*, size_t, size_t)`
17.5.2.1.50	lib.wstring::find.last.not.of.wc	`wstring::find_last_not_of(wchar_t, size_t)`
17.5.2.1.51	lib.wstring::substr	`wstring::substr(size_t, size_t)`

17.5.2.1.52	lib.wstring::compare.wsub	wstring::compare(const wstring&, size_t, size_t)
17.5.2.1.53	lib.wstring::compare.wstr	wstring::compare(const wchar_t*, size_t)
17.5.2.1.54	lib.wstring::compare.wc	wstring::compare(wchar_t, size_t)
17.5.2.2	lib.op+.wsub.wsub	operator+(const wstring&, const wstring&)
17.5.2.3	lib.op+.wstr.wsub	operator+(const wchar_t*, const wstring&)
17.5.2.4	lib.op+.wc.wsub	operator+(wchar_t, const wstring&)
17.5.2.5	lib.op+.wsub.wstr	operator+(const wstring&, const wchar_t*)
17.5.2.6	lib.op+.wsub.wc	operator+(const wstring&, wchar_t)
17.5.2.7	lib.op==.wsub.wsub	operator==(const wstring&, const wstring&)
17.5.2.8	lib.op==.wstr.wsub	operator==(const wchar_t*, const wstring&)
17.5.2.9	lib.op==.wc.wsub	operator==(wchar_t, const wstring&)
17.5.2.10	lib.op==.wsub.wstr	operator==(const wstring&, const wchar_t*)
17.5.2.11	lib.op==.wsub.wc	operator==(const wstring&, wchar_t)
17.5.2.12	lib.op!=.wsub.wsub	operator!=(const wstring&, const wstring&)
17.5.2.13	lib.op!=.wstr.wsub	operator!=(const wchar_t*, const wstring&)
17.5.2.14	lib.op!=.wc.wsub	operator!=(wchar_t, const wstring&)
17.5.2.15	lib.op!=.wsub.wstr	operator!=(const wstring&, const wchar_t*)
17.5.2.16	lib.op!=.wsub.wc	operator!=(const wstring&, wchar_t)
17.5.3	lib.header.bits	Header <bits>
17.5.3.1	lib.template.bits	Template class bits<*N*>
17.5.3.1.1	lib.cons.bits	bits<*N*>::bits()
17.5.3.1.2	lib.cons.bits.ul	bits<*N*>::bits(unsigned long)
17.5.3.1.3	lib.cons.bits.subt	bits<*N*>::bits(const string&, size_t, size_t)
17.5.3.1.4	lib.bits::op&=.bt	bits<*N*>::operator&=(const bits<*N*>&)
17.5.3.1.5	lib.bits::op|=.bt	bits<*N*>::operator|=(const bits<*N*>&)
17.5.3.1.6	lib.bits::op^=.bt	bits<*N*>::operator^=(const bits<*N*>&)
17.5.3.1.7	lib.bits::op.lsh=	bits<*N*>::operator<<=(size_t)
17.5.3.1.8	lib.bits::op.rsh=	bits<*N*>::operator>>=(size_t)
17.5.3.1.9	lib.bits::set	bits<*N*>::set()
17.5.3.1.10	lib.bits::set.n	bits<*N*>::set(size_t, int)
17.5.3.1.11	lib.bits::reset	bits<*N*>::reset()
17.5.3.1.12	lib.bits::reset.n	bits<*N*>::reset(size_t)
17.5.3.1.13	lib.bits::op~	bits<*N*>::operator~()
17.5.3.1.14	lib.bits::toggle	bits<*N*>::toggle()
17.5.3.1.15	lib.bits::toggle.n	bits<*N*>::toggle(size_t)
17.5.3.1.16	lib.bits::to.ushort	bits<*N*>::to_ushort()
17.5.3.1.17	lib.bits::to.ulong	bits<*N*>::to_ulong()
17.5.3.1.18	lib.bits::to.string	bits<*N*>::to_string()
17.5.3.1.19	lib.bits::count	bits<*N*>::count()
17.5.3.1.20	lib.bits::length	bits<*N*>::length()
17.5.3.1.21	lib.bits::op==.bt	bits<*N*>::operator==(const bits<*N*>&)
17.5.3.1.22	lib.bits::op!=.bt	bits<*N*>::operator!=(const bits<*N*>&)
17.5.3.1.23	lib.bits::test	bits<*N*>::test(size_t)

17.5.3.1.24	lib.bits::any	bits<*N*>::any()
17.5.3.1.25	lib.bits::none	bits<*N*>::none()
17.5.3.1.26	lib.bits::op.lsh	bits<*N*>::operator<<(size_t)
17.5.3.1.27	lib.bits::op.rsh	bits<*N*>::operator>>(size_t)
17.5.3.2	lib.op&.bt.bt	operator&(const bits<*N*>&, const bits<*N*>&)
17.5.3.3	lib.op|.bt.bt	operator|(const bits<*N*>&, const bits<*N*>&)
17.5.3.4	lib.op^.bt.bt	operator^(const bits<*N*>&, const bits<*N*>&)
17.5.3.5	lib.ext.bt	operator>>(istream&, bits<*N*>&)
17.5.3.6	lib.ins.bt	operator<<(ostream&, const bits<*N*>&)
17.5.4	lib.header.bitstring	Header <bitstring>
17.5.4.1	lib.bitstring	Class bitstring
17.5.4.1.1	lib.cons.bitstring	bitstring::bitstring()
17.5.4.1.2	lib.cons.bitstring.ul	bitstring::bitstring(unsigned long, size_t)
17.5.4.1.3	lib.cons.bitstring.bs	bitstring::bitstring(const bitstring&, size_t, size_t)
17.5.4.1.4	lib.cons.bitstring.sub	bitstring::bitstring(const string&, size_t, size_t)
17.5.4.1.5	lib.bitstring::op+=.bs	bitstring::operator+=(const bitstring&)
17.5.4.1.6	lib.bitstring::op&=.bs	bitstring::operator&=(const bitstring&)
17.5.4.1.7	lib.bitstring::op|=.bs	bitstring::operator|=(const bitstring&)
17.5.4.1.8	lib.bitstring::op^=.bs	bitstring::operator^=(const bitstring&)
17.5.4.1.9	lib.bitstring::op.lsh=	bitstring::operator<<=(size_t)
17.5.4.1.10	lib.bitstring::op.rsh=	bitstring::operator>>=(size_t)
17.5.4.1.11	lib.bitstring::append	bitstring::append(const bitstring&, size_t, size_t)
17.5.4.1.12	lib.bitstring::assign	bitstring::assign(const bitstring&, size_t, size_t)
17.5.4.1.13	lib.bitstring::insert	bitstring::insert(size_t, const bitstring&, size_t, size_t)
17.5.4.1.14	lib.bitstring::remove	bitstring::remove(size_t, size_t)
17.5.4.1.15	lib.bitstring::replace	bitstring::replace(size_t, size_t, const bitstring&, size_t, size_t)
17.5.4.1.16	lib.bitstring::set	bitstring::set()
17.5.4.1.17	lib.bitstring::set.n	bitstring::set(size_t, int)
17.5.4.1.18	lib.bitstring::reset	bitstring::reset()
17.5.4.1.19	lib.bitstring::reset.n	bitstring::reset(size_t)
17.5.4.1.20	lib.bitstring::op˜	bitstring::operator~()
17.5.4.1.21	lib.bitstring::toggle	bitstring::toggle()
17.5.4.1.22	lib.bitstring::toggle.n	bitstring::toggle(size_t)
17.5.4.1.23	lib.bitstring::to.string	bitstring::to_string()
17.5.4.1.24	lib.bitstring::count	bitstring::count()
17.5.4.1.25	lib.bitstring::length	bitstring::length()
17.5.4.1.26	lib.bitstring::resize	bitstring::resize(size_t, int)
17.5.4.1.27	lib.bitstring::trim	bitstring::trim()
17.5.4.1.28	lib.bitstring::find	bitstring::find(int, size_t, size_t)
17.5.4.1.29	lib.bitstring::rfind	bitstring::rfind(int, size_t, size_t)
17.5.4.1.30	lib.bitstring::substr	bitstring::substr(size_t, size_t)
17.5.4.1.31	lib.bitstring::op==.bs	bitstring::operator==(const bitstring&)
17.5.4.1.32	lib.bitstring::op!=.bs	bitstring::operator!=(const bitstring&)
17.5.4.1.33	lib.bitstring::test	bitstring::test(size_t)

17.5.4.1.34	lib.bitstring::any	bitstring::any()
17.5.4.1.35	lib.bitstring::none	bitstring::none()
17.5.4.1.36	lib.bitstring::op.lsh	bitstring::operator<<(size_t)
17.5.4.1.37	lib.bitstring::op.rsh	bitstring::operator>>(size_t)
17.5.4.2	lib.op+.bs.bs	operator+(const bitstring&, const bitstring&)
17.5.4.3	lib.op&.bs.bs	operator&(const bitstring&, const bitstring&)
17.5.4.4	lib.op|.bs.bs	operator|(const bitstring&, const bitstring&)
17.5.4.5	lib.op^.bs.bs	operator^(const bitstring&, const bitstring&)
17.5.4.6	lib.ext.bs	operator>>(istream&, bitstring&)
17.5.4.7	lib.ins.bs	operator<<(ostream&, const bitstring&)
17.5.5	lib.header.dynarray	Header <dynarray>
17.5.5.1	lib.template.dynarray	Template class dynarray<*T*>
17.5.5.1.1	lib.cons.dynarray	dynarray<*T*>::dynarray()
17.5.5.1.2	lib.cons.dynarray.cap	dynarray<*T*>::dynarray(size_t, capacity)
17.5.5.1.3	lib.cons.dynarray.da	dynarray<*T*>::dynarray(const dynarray<*T*>&)
17.5.5.1.4	lib.cons.dynarray.t	dynarray<*T*>::dynarray(const *T*&, size_t)
17.5.5.1.5	lib.cons.dynarray.pt	dynarray<*T*>::dynarray(const *T**, size_t)
17.5.5.1.6	lib.dynarray::op+=.da	dynarray<*T*>::operator+=(const dynarray<*T*>&)
17.5.5.1.7	lib.dynarray::op+=.t	dynarray<*T*>::operator+=(const *T*&)
17.5.5.1.8	lib.dynarray::append.t	dynarray<*T*>::append(const *T*&, size_t)
17.5.5.1.9	lib.dynarray::append.pt	dynarray<*T*>::append(const *T**, size_t)
17.5.5.1.10	lib.dynarray::assign.t	dynarray<*T*>::assign(const *T*&, size_t)
17.5.5.1.11	lib.dynarray::assign.pt	dynarray<*T*>::assign(const *T**, size_t)
17.5.5.1.12	lib.dynarray::insert.da	dynarray<*T*>::insert(size_t, const dynarray<*T*>&)
17.5.5.1.13	lib.dynarray::insert.t	dynarray<*T*>::insert(size_t, const *T*&, size_t)
17.5.5.1.14	lib.dynarray::insert.pt	dynarray<*T*>::insert(size_t, const *T**, size_t)
17.5.5.1.15	lib.dynarray::remove	dynarray<*T*>::remove(size_t, size_t)
17.5.5.1.16	lib.dynarray::swap	dynarray<*T*>::swap(dynarray<*T*>&)
17.5.5.1.17	lib.dynarray::sub.array	dynarray<*T*>::sub_array(dynarray<*T*>&, size_t, size_t)
17.5.5.1.18	lib.dynarray::get.at	dynarray<*T*>::get_at(size_t)
17.5.5.1.19	lib.dynarray::put.at	dynarray<*T*>::put_at(size_t, const *T*&)
17.5.5.1.20	lib.dynarray::op.array	dynarray<*T*>::operator[](size_t)
17.5.5.1.21	lib.dynarray::base	dynarray<*T*>::base()
17.5.5.1.22	lib.dynarray::length	dynarray<*T*>::length()
17.5.5.1.23	lib.dynarray::resize	dynarray<*T*>::resize(size_t)
17.5.5.1.24	lib.dynarray::resize.t	dynarray<*T*>::resize(size_t, const *T*&)
17.5.5.1.25	lib.dynarray::reserve	dynarray<*T*>::reserve()

17.5.5.1.26	lib.dynarray::reserve.cap	dynarray<T>::reserve(size_t)
17.5.5.2	lib.op+.da.da	operator+(const dynarray<T>&, const dynarray<T>&)
17.5.5.3	lib.op+.da.t	operator+(const dynarray<T>&, const T&)
17.5.5.4	lib.op+.t.da	operator+(const T&, const dynarray<T>&)
17.5.6	lib.header.ptrdynarray	Header <ptrdynarray>
17.5.6.1	lib.template.ptrdynarray	Template class ptrdynarray<T>
17.5.6.1.1	lib.cons.ptrdynarray	ptrdynarray<T>::ptrdynarray()
17.5.6.1.2	lib.cons.ptrdynarray.cap	ptrdynarray<T>::ptrdynarray(size_t, capacity)
17.5.6.1.3	lib.cons.ptrdynarray.pda	ptrdynarray<T>::ptrdynarray(const ptrdynarray<T>&)
17.5.6.1.4	lib.cons.ptrdynarray.pt	ptrdynarray<T>::ptrdynarray(T*)
17.5.6.1.5	lib.cons.ptrdynarray.ppt	ptrdynarray<T>::ptrdynarray(const T**, size_t)
17.5.6.1.6	lib.ptrdynarray::op+=.pda	ptrdynarray<T>::operator+=(const ptrdynarray<T>&)
17.5.6.1.7	lib.ptrdynarray::op+=.pt	ptrdynarray<T>::operator+=(T*)
17.5.6.1.8	lib.ptrdynarray::append.pt	ptrdynarray<T>::append(T*, size_t)
17.5.6.1.9	lib.ptrdynarray::append.ppt	ptrdynarray<T>::append(T**, size_t)
17.5.6.1.10	lib.ptrdynarray::assign.pt	ptrdynarray<T>::assign(T*, size_t)
17.5.6.1.11	lib.ptrdynarray::assign.ppt	ptrdynarray<T>::assign(T**, size_t)
17.5.6.1.12	lib.ptrdynarray::insert.pda	ptrdynarray<T>::insert(size_t, const ptrdynarray<T>&, size_t)
17.5.6.1.13	lib.ptrdynarray::insert.pt	ptrdynarray<T>::insert(size_t, T*, size_t)
17.5.6.1.14	lib.ptrdynarray::insert.ppt	ptrdynarray<T>::insert(size_t, T**, size_t)
17.5.6.1.15	lib.ptrdynarray::remove	ptrdynarray<T>::remove(size_t, size_t)
17.5.6.1.16	lib.ptrdynarray::swap	ptrdynarray<T>::swap(ptrdynarray<T>&)
17.5.6.1.17	lib.ptrdynarray::sub.array	ptrdynarray<T>::sub_array(ptrdynarray<T>&, size_t, size_t)
17.5.6.1.18	lib.ptrdynarray::get.at	ptrdynarray<T>::get_at(size_t)
17.5.6.1.19	lib.ptrdynarray::put.at	ptrdynarray<T>::put_at(size_t, const T&)
17.5.6.1.20	lib.ptrdynarray::op.array	ptrdynarray<T>::operator[](size_t)
17.5.6.1.21	lib.ptrdynarray::base	ptrdynarray<T>::base()
17.5.6.1.22	lib.ptrdynarray::length	ptrdynarray<T>::length()
17.5.6.1.23	lib.ptrdynarray::resize	ptrdynarray<T>::resize(size_t)
17.5.6.1.24	lib.ptrdynarray::resize.pt	ptrdynarray<T>::resize(size_t, T*)
17.5.6.1.25	lib.ptrdynarray::reserve	ptrdynarray<T>::reserve()
17.5.6.1.26	lib.ptrdynarray::reserve.cap	ptrdynarray<T>::reserve(size_t)
17.5.6.2	lib.op+.pda.pda	operator+(const ptrdynarray<T>&, const ptrdynarray<T>&)
17.5.6.3	lib.op+.pda.pt	operator+(const ptrdynarray<T>&, T*)
17.5.6.4	lib.op+.pt.pda	operator+(T*, const ptrdynarray<T>&)
17.5.7	lib.header.complex	Header <complex>
17.5.7.1	lib.complex.with.float	Complex numbers with float precision
17.5.7.1.1	lib.float.complex	Class float_complex
17.5.7.1.1.1	lib.cons.float.complex.f.f	float_complex::float_complex(float, float)
17.5.7.1.1.2	lib.op+=.fc	operator+=(float_complex)
17.5.7.1.1.3	lib.op-=.fc	operator-=(float_complex)
17.5.7.1.1.4	lib.op*=.fc	operator*=(float_complex)

17.5.7.1.1.5	lib.op/=.fc	operator/=(float_complex)
17.5.7.1.2	lib..float.complex.dc	_float_complex(const double_complex&)
17.5.7.1.3	lib..float.complex.ldc	_float_complex(const long_double_complex&)
17.5.7.1.4	lib.op+.fc.fc	operator+(float_complex, float_complex)
17.5.7.1.5	lib.op+.fc.f	operator+(float_complex, float)
17.5.7.1.6	lib.op+.f.fc	operator+(float, float_complex)
17.5.7.1.7	lib.op-.fc.fc	operator-(float_complex, float_complex)
17.5.7.1.8	lib.op-.fc.f	operator-(float_complex, float)
17.5.7.1.9	lib.op-.f.fc	operator-(float, float_complex)
17.5.7.1.10	lib.op*.fc.fc	operator*(float_complex, float_complex)
17.5.7.1.11	lib.op*.fc.f	operator*(float_complex, float)
17.5.7.1.12	lib.op*.f.fc	operator*(float, float_complex)
17.5.7.1.13	lib.op/.fc.fc	operator/(float_complex, float_complex)
17.5.7.1.14	lib.op/.fc.f	operator/(float_complex, float)
17.5.7.1.15	lib.op/.f.fc	operator/(float, float_complex)
17.5.7.1.16	lib.op+.fc	operator+(float_complex)
17.5.7.1.17	lib.op-.fc	operator-(float_complex)
17.5.7.1.18	lib.op==.fc.fc	operator==(float_complex, float_complex)
17.5.7.1.19	lib.op==.fc.f	operator==(float_complex, float)
17.5.7.1.20	lib.op==.f.fc	operator==(float, float_complex)
17.5.7.1.21	lib.op!=.fc.fc	operator!=(float_complex, float_complex)
17.5.7.1.22	lib.op!=.fc.f	operator!=(float_complex, float)
17.5.7.1.23	lib.op!=.f.fc	operator!=(float, float_complex)
17.5.7.1.24	lib.ext.fc	operator>>(istream&, float_complex&)
17.5.7.1.25	lib.ins.fc	operator<<(ostream&, float_complex)
17.5.7.1.26	lib.abs.fc	abs(float_complex)
17.5.7.1.27	lib.arg.fc	arg(float_complex)
17.5.7.1.28	lib.conj.fc	conj(float_complex)
17.5.7.1.29	lib.cos.fc	cos(float_complex)
17.5.7.1.30	lib.cosh.fc	cosh(float_complex)
17.5.7.1.31	lib.exp.fc	exp(float_complex)
17.5.7.1.32	lib.imag.fc	imag(float_complex)
17.5.7.1.33	lib.log.fc	log(float_complex)
17.5.7.1.34	lib.norm.fc	norm(float_complex)
17.5.7.1.35	lib.polar.f.f	polar(float, float)
17.5.7.1.36	lib.pow.fc.fc	pow(float_complex, float_complex)
17.5.7.1.37	lib.pow.fc.f	pow(float_complex, float)
17.5.7.1.38	lib.pow.fc.i	pow(float_complex, int)
17.5.7.1.39	lib.pow.f.fc	pow(float, float_complex)
17.5.7.1.40	lib.real.fc	real(float_complex)
17.5.7.1.41	lib.sin.fc	sin(float_complex)
17.5.7.1.42	lib.sinh.fc	sinh(float_complex)
17.5.7.1.43	lib.sqrt.fc	sqrt(float_complex)
17.5.7.2	lib.complex.with.d	Complex numbers with double precision
17.5.7.2.1	lib.double.complex	Class double_complex
17.5.7.2.1.1	lib.cons.double.complex.d.d	double_complex::double_complex(double, double)
17.5.7.2.1.2	lib.cons.double.complex.fc	double_complex::double_complex(float_complex&)
17.5.7.2.1.3	lib.op+=.dc	operator+=(double_complex)

17.5.7.2.1.4	lib.op-=.dc	`operator-=(double_complex)`
17.5.7.2.1.5	lib.op*=.dc	`operator*=(double_complex)`
17.5.7.2.1.6	lib.op/=.dc	`operator/=(double_complex)`
17.5.7.2.2	lib..double.complex.ldc	`_double_complex(const long_double_complex&)`
17.5.7.2.3	lib.op+.dc.dc	`operator+(double_complex, double_complex)`
17.5.7.2.4	lib.op+.dc.d	`operator+(double_complex, double)`
17.5.7.2.5	lib.op+.d.dc	`operator+(double, double_complex)`
17.5.7.2.6	lib.op-.dc.dc	`operator-(double_complex, double_complex)`
17.5.7.2.7	lib.op-.dc.d	`operator-(double_complex, double)`
17.5.7.2.8	lib.op-.d.dc	`operator-(double, double_complex)`
17.5.7.2.9	lib.op*.dc.dc	`operator*(double_complex, double_complex)`
17.5.7.2.10	lib.op*.dc.d	`operator*(double_complex, double)`
17.5.7.2.11	lib.op*.d.dc	`operator*(double, double_complex)`
17.5.7.2.12	lib.op/.dc.dc	`operator/(double_complex, double_complex)`
17.5.7.2.13	lib.op/.dc.d	`operator/(double_complex, double)`
17.5.7.2.14	lib.op/.d.dc	`operator/(double, double_complex)`
17.5.7.2.15	lib.op+.dc	`operator+(double_complex)`
17.5.7.2.16	lib.op-.dc	`operator-(double_complex)`
17.5.7.2.17	lib.op==.dc.dc	`operator==(double_complex, double_complex)`
17.5.7.2.18	lib.op==.dc.d	`operator==(double_complex, double)`
17.5.7.2.19	lib.op==.d.dc	`operator==(double, double_complex)`
17.5.7.2.20	lib.op!=.dc.dc	`operator!=(double_complex, double_complex)`
17.5.7.2.21	lib.op!=.dc.d	`operator!=(double_complex, double)`
17.5.7.2.22	lib.op!=.d.dc	`operator!=(double, double_complex)`
17.5.7.2.23	lib.ext.dc	`operator>>(istream&, double_complex&)`
17.5.7.2.24	lib.ins.dc	`operator<<(ostream&, double_complex)`
17.5.7.2.25	lib.abs.dc	`abs(double_complex)`
17.5.7.2.26	lib.arg.dc	`arg(double_complex)`
17.5.7.2.27	lib.conj.dc	`conj(double_complex)`
17.5.7.2.28	lib.cos.dc	`cos(double_complex)`
17.5.7.2.29	lib.cosh.dc	`cosh(double_complex)`
17.5.7.2.30	lib.exp.dc	`exp(double_complex)`
17.5.7.2.31	lib.imag.dc	`imag(double_complex)`
17.5.7.2.32	lib.log.dc	`log(double_complex)`
17.5.7.2.33	lib.norm.dc	`norm(double_complex)`
17.5.7.2.34	lib.polar.d.d	`polar(double, double)`
17.5.7.2.35	lib.pow.dc.dc	`pow(double_complex, double_complex)`
17.5.7.2.36	lib.pow.dc.d	`pow(double_complex, double)`
17.5.7.2.37	lib.pow.dc.i	`pow(double_complex, int)`
17.5.7.2.38	lib.pow.d.dc	`pow(double, double_complex)`
17.5.7.2.39	lib.real.dc	`real(double_complex)`
17.5.7.2.40	lib.sin.dc	`sin(double_complex)`
17.5.7.2.41	lib.sinh.dc	`sinh(double_complex)`
17.5.7.2.42	lib.sqrt.dc	`sqrt(double_complex)`
17.5.7.3	lib.complex.with.ld	Complex numbers with `long double` precision
17.5.7.3.1	lib.long.double.complex	Class `long_double_complex`
17.5.7.3.1.1	lib.cons.long.double.complex.ld.ld	

		`long_double_complex::long_double_complex(long double, long double)`
17.5.7.3.1.2	lib.cons.long.double.complex.fc	`long_double_complex::long_double_complex (float_complex&)`
17.5.7.3.1.3	lib.cons.long.double.complex.dc	`long_double_complex::long_double_complex (double_complex&)`
17.5.7.3.1.4	lib.op+=.ldc	`operator+=(long_double_complex)`
17.5.7.3.1.5	lib.op-=.ldc	`operator-=(long_double_complex)`
17.5.7.3.1.6	lib.op*=.ldc	`operator*=(long_double_complex)`
17.5.7.3.1.7	lib.op/=.ldc	`operator/=(long_double_complex)`
17.5.7.3.2	lib.op+.ldc.ldc	`operator+(long_double_complex, long_double_complex)`
17.5.7.3.3	lib.op+.ldc.ld	`operator+(long_double_complex, long double)`
17.5.7.3.4	lib.op+.ld.ldc	`operator+(long double, long_double_complex)`
17.5.7.3.5	lib.op-.ldc.ldc	`operator-(long_double_complex, long_double_complex)`
17.5.7.3.6	lib.op-.ldc.ld	`operator-(long_double_complex, long double)`
17.5.7.3.7	lib.op-.ld.ldc	`operator-(long double, long_double_complex)`
17.5.7.3.8	lib.op*.ldc.ldc	`operator*(long_double_complex, long_double_complex)`
17.5.7.3.9	lib.op*.ldc.ld	`operator*(long_double_complex, long double)`
17.5.7.3.10	lib.op*.ld.ldc	`operator*(long double, long_double_complex)`
17.5.7.3.11	lib.op/.ldc.ldc	`operator/(long_double_complex, long_double_complex)`
17.5.7.3.12	lib.op/.ldc.ld	`operator/(long_double_complex, long double)`
17.5.7.3.13	lib.op/.ld.ldc	`operator/(long double, long_double_complex)`
17.5.7.3.14	lib.op+.ldc	`operator+(long_double_complex)`
17.5.7.3.15	lib.op-.ldc	`operator-(long_double_complex)`
17.5.7.3.16	lib.op==.ldc.ldc	`operator==(long_double_complex, long_double_complex)`
17.5.7.3.17	lib.op==.ldc.ld	`operator==(long_double_complex, long double)`
17.5.7.3.18	lib.op==.ld.ldc	`operator==(long double, long_double_complex)`
17.5.7.3.19	lib.op!=.ldc.ldc	`operator!=(long_double_complex, long_double_complex)`
17.5.7.3.20	lib.op!=.ldc.ld	`operator!=(long_double_complex, long double)`
17.5.7.3.21	lib.op!=.ld.ldc	`operator!=(long double, long_double_complex)`
17.5.7.3.22	lib.ext.ldc	`operator>>(istream&, long_double_complex&)`
17.5.7.3.23	lib.ins.ldc	`operator<<(ostream&, long_double_complex)`
17.5.7.3.24	lib.abs.ldc	`abs(long_double_complex)`

17.5.7.3.25	lib.arg.ldc	`arg(long_double_complex)`
17.5.7.3.26	lib.conj.ldc	`conj(long_double_complex)`
17.5.7.3.27	lib.cos.ldc	`cos(long_double_complex)`
17.5.7.3.28	lib.cosh.ldc	`cosh(long_double_complex)`
17.5.7.3.29	lib.exp.ldc	`exp(long_double_complex)`
17.5.7.3.30	lib.imag.ldc	`imag(long_double_complex)`
17.5.7.3.31	lib.log.ldc	`log(long_double_complex)`
17.5.7.3.32	lib.norm.ldc	`norm(long_double_complex)`
17.5.7.3.33	lib.polar.ld.ld	`polar(long double, long double)`
17.5.7.3.34	lib.pow.ldc.ldc	`pow(long_double_complex, long_double_complex)`
17.5.7.3.35	lib.pow.ldc.ld	`pow(long_double_complex, long double)`
17.5.7.3.36	lib.pow.ldc.i	`pow(long_double_complex, int)`
17.5.7.3.37	lib.pow.ld.ldc	`pow(long double, long_double_complex)`
17.5.7.3.38	lib.real.ldc	`real(long_double_complex)`
17.5.7.3.39	lib.sin.ldc	`sin(long_double_complex)`
17.5.7.3.40	lib.sinh.ldc	`sinh(long_double_complex)`
17.5.7.3.41	lib.sqrt.ldc	`sqrt(long_double_complex)`
A	gram	Grammar summary
A.1	gram.key	Keywords
A.2	gram.lex	Lexical conventions
A.3	gram.basic	Basic concepts
A.4	gram.expr	Expressions
A.5	gram.stmt.stmt	Statements
A.6	gram.dcl.dcl	Declarations
A.7	gram.dcl.decl	Declarators
A.8	gram.class	Classes
A.9	gram.class.derived	Derived classes
A.10	gram.special	Special member functions
A.11	gram.over	Overloading
A.12	gram.temp	Templates
A.13	gram.except	Exception handling
B	limits	Implementation quantities
C	diff	Compatibility
C.1	diff.c	Extensions
C.1.1	diff.early	C++ features available in 1985
C.1.2	diff.c++	C++ features added since 1985
C.2	diff.iso	C++ and ISO C
C.2.1	diff.lex	Clause _lex_: lexical conventions
C.2.2	diff.basic	Clause _basic_: basic concepts
C.2.3	diff.expr	Clause _expr_: expressions
C.2.4	diff.stat	Clause _stmt.stmt_: statements
C.2.5	diff.dcl	Clause _dcl.dcl_: declarations
C.2.6	diff.decl	Clause _dcl.decl_: declarators
C.2.7	diff.class	Clause _class_: classes
C.2.8	diff.cpp	Clause _cpp_: preprocessing directives
C.3	diff.anac	Anachronisms
C.3.1	diff.fct.def	Old style function definitions
C.3.2	diff.base.init	Old style base class initializer
C.3.3	diff.this	Assignment to `this`
C.3.4	diff.bound	Cast of bound pointer
C.3.5	diff.class.nonnested	Nonnested classes
D	future.directions	Future directions

Listing by symbolic name

basic	3	Basic concepts
basic.compound	3.8.2	Compound types
basic.def	3.1	Declarations and definitions
basic.def.odr	3.2	One definition rule
basic.file.scope	3.3.4	File scope
basic.fundamental	3.8.1	Fundamental types
basic.link	3.4	Program and linkage
basic.lval	3.9	Lvalues and rvalues
basic.namespace	7.3	Namespaces
basic.scope	3.3	Declarative regions and scopes
basic.scope.class	3.3.6	Class scope
basic.scope.elab	3.3.9	Elaborated type specifier
basic.scope.exqual	3.3.8	Explicit qualification
basic.scope.hiding	3.3.7	Name hiding
basic.scope.local	3.3.1	Local scope
basic.scope.namespace	3.3.5	Namespace scope
basic.scope.pdecl	3.3.10	Point of declaration
basic.scope.proto	3.3.2	Function prototype scope
basic.start	3.5	Start and termination
basic.start.init	3.5.2	Initialization of non-local objects
basic.start.main	3.5.1	Main function
basic.start.term	3.6	Termination
basic.stc	3.7	Storage duration
basic.stc.auto	3.7.2	Automatic storage duration
basic.stc.dynamic	3.7.3	Dynamic storage class
basic.stc.inherit	3.7.4	Duration of sub-objects
basic.stc.mutable	3.7.5	The `mutable` keyword
basic.stc.ref	3.7.6	Reference duration
basic.stc.static	3.7.1	Static storage duration
basic.type.name	3.8.4	Type names
basic.type.qualifier	3.8.3	CV-qualifiers
basic.types	3.8	Types
class	9	Classes
class.abstract	10.4	Abstract classes
class.access	11	Member access control
class.access.base	11.2	Access specifiers for base classes
class.access.dcl	11.3	Access declarations
class.access.spec	11.1	Access specifiers
class.access.virt	11.6	Access to virtual functions
class.ambig	10.2	Ambiguities
class.base.init	12.6.2	Initializing bases and members
class.bit	9.7	Bit-fields
class.cdtor	12.7	Constructors and destructors
class.conv	12.3	Conversions
class.conv.ctor	12.3.1	Conversion by constructor
class.conv.fct	12.3.2	Conversion functions
class.copy	12.8	Copying class objects
class.ctor	12.1	Constructors
class.derived	10	Derived classes
class.dtor	12.4	Destructors
class.expl.init	12.6.1	Explicit initialization
class.free	12.5	Free store
class.friend	11.4	Friends

class.init	12.6	Initialization
class.inline	9.4.2	Inline member functions
class.local	9.9	Local class declarations
class.mem	9.2	Class members
class.mfct	9.4	Member functions
class.mi	10.1	Multiple base classes
class.name	9.1	Class names
class.nest	9.8	Nested class declarations
class.nested.type	9.10	Nested type names
class.paths	11.7	Multiple access
class.protected	11.5	Protected member access
class.scope	10.5	Summary of scope rules
class.scope0	9.3	Scope rules for classes
class.static	9.5	Static members
class.temporary	12.2	Temporary objects
class.this	9.4.1	The `this` pointer
class.union	9.6	Unions
class.virtual	10.3	Virtual functions
conv	4	Standard conversions
conv.arith	4.5	Arithmetic conversions
conv.bool	4.9	Boolean conversions
conv.double	4.3	Float and double
conv.float	4.4	Floating and integral
conv.integral	4.2	Integral conversions
conv.mem	4.8	Pointers to members
conv.prom	4.1	Integral promotions
conv.ptr	4.6	Pointer conversions
conv.ref	4.7	Reference conversions
cpp	16	Preprocessing directives
cpp.concat	16.3.3	The `##` operator
cpp.cond	16.1	Conditional inclusion
cpp.error	16.5	Error directive
cpp.include	16.2	Source file inclusion
cpp.line	16.4	Line control
cpp.null	16.7	Null directive
cpp.pragma	16.6	Pragma directive
cpp.predefined	16.8	Predefined macro names
cpp.replace	16.3	Macro replacement
cpp.rescan	16.3.4	Rescanning and further replacement
cpp.scope	16.3.5	Scope of macro definitions
cpp.stringize	16.3.2	The `#` operator
cpp.subst	16.3.1	Argument substitution
dcl.ambig.res	8.2	Ambiguity resolution
dcl.array	8.3.4	Arrays
dcl.asm	7.4	The `asm` declaration
dcl.dcl	7	Declarations
dcl.decl	8	Declarators
dcl.enum	7.2	Enumeration declarations
dcl.fct	8.3.5	Functions
dcl.fct.def	8.4	Function definitions
dcl.fct.default	8.3.6	Default arguments
dcl.fct.spec	7.1.2	Function specifiers
dcl.friend	7.1.4	The `friend` specifier
dcl.init	8.5	Initializers
dcl.init.aggr	8.5.1	Aggregates

dcl.init.ref	8.5.3	References
dcl.init.string	8.5.2	Character arrays
dcl.link	7.5	Linkage specifications
dcl.meaning	8.3	Meaning of declarators
dcl.mptr	8.3.3	Pointers to members
dcl.name	8.1	Type names
dcl.ptr	8.3.1	Pointers
dcl.ref	8.3.2	References
dcl.spec	7.1	Specifiers
dcl.stc	7.1.1	Storage class specifiers
dcl.type	7.1.5	Type specifiers
dcl.type.cv	7.1.5.1	The *cv-qualifiers*
dcl.type.elab	7.1.5.3	Elaborated type specifiers
dcl.type.simple	7.1.5.2	Simple type specifiers
dcl.typedef	7.1.3	The `typedef` specifier
diff	C	Compatibility
diff.anac	C.3	Anachronisms
diff.base.init	C.3.2	Old style base class initializer
diff.basic	C.2.2	Clause _basic_: basic concepts
diff.bound	C.3.4	Cast of bound pointer
diff.c	C.1	Extensions
diff.c++	C.1.2	C++ features added since 1985
diff.class	C.2.7	Clause _class_: classes
diff.class.nonnested	C.3.5	Nonnested classes
diff.cpp	C.2.8	Clause _cpp_: preprocessing directives
diff.dcl	C.2.5	Clause _dcl.dcl_: declarations
diff.decl	C.2.6	Clause _dcl.decl_: declarators
diff.early	C.1.1	C++ features available in 1985
diff.expr	C.2.3	Clause _expr_: expressions
diff.fct.def	C.3.1	Old style function definitions
diff.iso	C.2	C++ and ISO C
diff.lex	C.2.1	Clause _lex_: lexical conventions
diff.stat	C.2.4	Clause _stmt.stmt_: statements
diff.this	C.3.3	Assignment to `this`
except	15	Exception handling
except.access	15.7	Exceptions and access
except.ctor	15.3	Constructors and destructors
except.handle	15.4	Handling an exception
except.intro	15.1	Exception handling
except.spec	15.5	Exception specifications
except.special	15.6	Special functions
except.terminate	15.6.1	The `terminate()` function
except.throw	15.2	Throwing an exception
except.unexpected	15.6.2	The `unexpected()` function
expr	5	Expressions
expr.add	5.7	Additive operators
expr.ass	5.17	Assignment operators
expr.bit.and	5.11	Bitwise AND operator
expr.call	5.2.2	Function call
expr.cast	5.4	Explicit type conversion (cast notation)
expr.comma	5.18	Comma operator
expr.cond	5.16	Conditional operator
expr.const	5.19	Constant expressions
expr.const.cast	5.2.10	Const cast
expr.delete	5.3.5	Delete

expr.dynamic.cast	5.2.6	Dynamic cast
expr.eq	5.10	Equality operators
expr.log.and	5.14	Logical AND operator
expr.log.or	5.15	Logical OR operator
expr.mptr.oper	5.5	Pointer-to-member operators
expr.mul	5.6	Multiplicative operators
expr.new	5.3.4	New
expr.or	5.13	Bitwise inclusive OR operator
expr.post	5.2	Postfix expressions
expr.post.incr	5.2.5	Increment and decrement
expr.pre.incr	5.3.2	Increment and decrement
expr.prim	5.1	Primary expressions
expr.ref	5.2.4	Class member access
expr.reinterpret.cast	5.2.9	Reinterpret cast
expr.rel	5.9	Relational operators
expr.shift	5.8	Shift operators
expr.sizeof	5.3.3	Sizeof
expr.static.cast	5.2.8	Static cast
expr.sub	5.2.1	Subscripting
expr.type.conv	5.2.3	Explicit type conversion (functional notation)
expr.typeid	5.2.7	Type identification
expr.unary	5.3	Unary expressions
expr.unary.op	5.3.1	Unary operators
expr.xor	5.12	Bitwise exclusive OR operator
future.directions	D	Future directions
gram	A	Grammar summary
gram.basic	A.3	Basic concepts
gram.class	A.8	Classes
gram.class.derived	A.9	Derived classes
gram.dcl.dcl	A.6	Declarations
gram.dcl.decl	A.7	Declarators
gram.except	A.13	Exception handling
gram.expr	A.4	Expressions
gram.key	A.1	Keywords
gram.lex	A.2	Lexical conventions
gram.over	A.11	Overloading
gram.special	A.10	Special member functions
gram.stmt.stmt	A.5	Statements
gram.temp	A.12	Templates
intro	1	General
intro.compliance	1.6	Processor compliance
intro.defs	1.3	Definitions
intro.execution	1.7	Program execution
intro.memory	1.5	The C++ memory model
intro.refs	1.2	Normative references
intro.scope	1.1	Scope
lex	2	Lexical conventions
lex.bool	2.9.5	Boolean literals
lex.ccon	2.9.2	Character literals
lex.comment	2.6	Comments
lex.digraph	2.4	Digraph sequences
lex.fcon	2.9.3	Floating literals
lex.icon	2.9.1	Integer literals
lex.key	2.8	Keywords
lex.literal	2.9	Literals

lex.name 2.7 Identifiers
lex.phases 2.1 Phases of translation
lex.pptoken 2.3 Preprocessing tokens
lex.string 2.9.4 String literals
lex.token 2.5 Tokens
lex.trigraph 2.2 Trigraph sequences
lib..cons.wstring.wc 17.5.2.1.5 `wstring::wstring(wchar_t, size_t)`
lib..cons.wstring.wstr 17.5.2.1.4 `wstring::wstring(const wchar_t*, size_t)`
lib..double.complex.ldc 17.5.7.2.2 `_double_complex(const long_double_complex&)`
lib..float.complex.dc 17.5.7.1.2 `_float_complex(const double_complex&)`
lib..float.complex.ldc 17.5.7.1.3 `_float_complex(const long_double_complex&)`
lib.abs.dc 17.5.7.2.25 `abs(double_complex)`
lib.abs.fc 17.5.7.1.26 `abs(float_complex)`
lib.abs.ldc 17.5.7.3.24 `abs(long_double_complex)`
lib.alternate.definitions.for.functions 17.1.5.4 Alternate definitions for functions
lib.arg.dc 17.5.7.2.26 `arg(double_complex)`
lib.arg.fc 17.5.7.1.27 `arg(float_complex)`
lib.arg.ldc 17.5.7.3.25 `arg(long_double_complex)`
lib.badcast 17.3.2.4 Class `badcast`
lib.badcast::do.raise 17.3.2.4.3 `badcast::do_raise()`
lib.badtypeid 17.3.4.1 Class `badtypeid`
lib.badtypeid::do.raise 17.3.4.1.3 `badtypeid::do_raise()`
lib.bitmask.types 17.1.5.10.2 Bitmask types
lib.bits::any 17.5.3.1.24 `bits<N>::any()`
lib.bits::count 17.5.3.1.19 `bits<N>::count()`
lib.bits::length 17.5.3.1.20 `bits<N>::length()`
lib.bits::none 17.5.3.1.25 `bits<N>::none()`
lib.bits::op!=.bt 17.5.3.1.22 `bits<N>::operator!=(const bits<N>&)`
lib.bits::op&=.bt 17.5.3.1.4 `bits<N>::operator&=(const bits<N>&)`
lib.bits::op.lsh 17.5.3.1.26 `bits<N>::operator<<(size_t)`
lib.bits::op.lsh= 17.5.3.1.7 `bits<N>::operator<<=(size_t)`
lib.bits::op.rsh 17.5.3.1.27 `bits<N>::operator>>(size_t)`
lib.bits::op.rsh= 17.5.3.1.8 `bits<N>::operator>>=(size_t)`
lib.bits::op==.bt 17.5.3.1.21 `bits<N>::operator==(const bits<N>&)`
lib.bits::op^=.bt 17.5.3.1.6 `bits<N>::operator^=(const bits<N>&)`
lib.bits::op|=.bt 17.5.3.1.5 `bits<N>::operator|=(const bits<N>&)`
lib.bits::op~ 17.5.3.1.13 `bits<N>::operator~()`
lib.bits::reset 17.5.3.1.11 `bits<N>::reset()`
lib.bits::reset.n 17.5.3.1.12 `bits<N>::reset(size_t)`
lib.bits::set 17.5.3.1.9 `bits<N>::set()`
lib.bits::set.n 17.5.3.1.10 `bits<N>::set(size_t, int)`
lib.bits::test 17.5.3.1.23 `bits<N>::test(size_t)`
lib.bits::to.string 17.5.3.1.18 `bits<N>::to_string()`
lib.bits::to.ulong 17.5.3.1.17 `bits<N>::to_ulong()`
lib.bits::to.ushort 17.5.3.1.16 `bits<N>::to_ushort()`
lib.bits::toggle 17.5.3.1.14 `bits<N>::toggle()`
lib.bits::toggle.n 17.5.3.1.15 `bits<N>::toggle(size_t)`
lib.bitstring 17.5.4.1 Class `bitstring`
lib.bitstring::any 17.5.4.1.34 `bitstring::any()`
lib.bitstring::append 17.5.4.1.11 `bitstring::append(const bitstring&, size_t, size_t)`
lib.bitstring::assign 17.5.4.1.12 `bitstring::assign(const bitstring&,`

		`size_t, size_t)`
lib.bitstring::count	17.5.4.1.24	`bitstring::count()`
lib.bitstring::find	17.5.4.1.28	`bitstring::find(int, size_t, size_t)`
lib.bitstring::insert	17.5.4.1.13	`bitstring::insert(size_t,`
		`const bitstring&, size_t, size_t)`
lib.bitstring::length	17.5.4.1.25	`bitstring::length()`
lib.bitstring::none	17.5.4.1.35	`bitstring::none()`
lib.bitstring::op!=.bs	17.5.4.1.32	`bitstring::operator!=(const bitstring&)`
lib.bitstring::op&=.bs	17.5.4.1.6	`bitstring::operator&=(const bitstring&)`
lib.bitstring::op+=.bs	17.5.4.1.5	`bitstring::operator+=(const bitstring&)`
lib.bitstring::op.lsh	17.5.4.1.36	`bitstring::operator<<(size_t)`
lib.bitstring::op.lsh=	17.5.4.1.9	`bitstring::operator<<=(size_t)`
lib.bitstring::op.rsh	17.5.4.1.37	`bitstring::operator>>(size_t)`
lib.bitstring::op.rsh=	17.5.4.1.10	`bitstring::operator>>=(size_t)`
lib.bitstring::op==.bs	17.5.4.1.31	`bitstring::operator==(const bitstring&)`
lib.bitstring::op^=.bs	17.5.4.1.8	`bitstring::operator^=(const bitstring&)`
lib.bitstring::op|=.bs	17.5.4.1.7	`bitstring::operator|=(const bitstring&)`
lib.bitstring::op˜	17.5.4.1.20	`bitstring::operator~()`
lib.bitstring::remove	17.5.4.1.14	`bitstring::remove(size_t, size_t)`
lib.bitstring::replace	17.5.4.1.15	`bitstring::replace(size_t, size_t,`
		`const bitstring&, size_t, size_t)`
lib.bitstring::reset	17.5.4.1.18	`bitstring::reset()`
lib.bitstring::reset.n	17.5.4.1.19	`bitstring::reset(size_t)`
lib.bitstring::resize	17.5.4.1.26	`bitstring::resize(size_t, int)`
lib.bitstring::rfind	17.5.4.1.29	`bitstring::rfind(int, size_t,`
		`size_t)`
lib.bitstring::set	17.5.4.1.16	`bitstring::set()`
lib.bitstring::set.n	17.5.4.1.17	`bitstring::set(size_t, int)`
lib.bitstring::substr	17.5.4.1.30	`bitstring::substr(size_t, size_t)`
lib.bitstring::test	17.5.4.1.33	`bitstring::test(size_t)`
lib.bitstring::to.string	17.5.4.1.23	`bitstring::to_string()`
lib.bitstring::toggle	17.5.4.1.21	`bitstring::toggle()`
lib.bitstring::toggle.n	17.5.4.1.22	`bitstring::toggle(size_t)`
lib.bitstring::trim	17.5.4.1.27	`bitstring::trim()`
lib.capacity	17.3.1.5	Type `capacity`
lib.cerr	17.4.9.3	Object `cerr`
lib.cin	17.4.9.1	Object `cin`
lib.clog	17.4.9.4	Object `clog`
lib.complex.with.d	17.5.7.2	Complex numbers with `double` precision
lib.complex.with.float	17.5.7.1	Complex numbers with `float` precision
lib.complex.with.ld	17.5.7.3	Complex numbers with `long double`
		precision
lib.conj.dc	17.5.7.2.27	`conj(double_complex)`
lib.conj.fc	17.5.7.1.28	`conj(float_complex)`
lib.conj.ldc	17.5.7.3.26	`conj(long_double_complex)`
lib.cons.badcast	17.3.2.4.1	`badcast::badcast()`
lib.cons.badtypeid	17.3.4.1.1	`badtypeid::badtypeid()`
lib.cons.bits	17.5.3.1.1	`bits<N>::bits()`
lib.cons.bits.subt	17.5.3.1.3	`bits<N>::bits(const string&, size_t,`
		`size_t)`
lib.cons.bits.ul	17.5.3.1.2	`bits<N>::bits(unsigned long)`
lib.cons.bitstring	17.5.4.1.1	`bitstring::bitstring()`
lib.cons.bitstring.bs	17.5.4.1.3	`bitstring::bitstring(const bitstring&,`
		`size_t, size_t)`
lib.cons.bitstring.sub	17.5.4.1.4	`bitstring::bitstring(const string&,`

		size_t, size_t)
lib.cons.bitstring.ul	17.5.4.1.2	bitstring::bitstring(unsigned long, size_t)
lib.cons.double.complex.d.d	17.5.7.2.1.1	double_complex::double_complex(double, double)
lib.cons.double.complex.fc	17.5.7.2.1.2	double_complex::double_complex(float_complex&)
lib.cons.dynarray	17.5.5.1.1	dynarray<*T*>::dynarray()
lib.cons.dynarray.cap	17.5.5.1.2	dynarray<*T*>::dynarray(size_t, capacity)
lib.cons.dynarray.da	17.5.5.1.3	dynarray<*T*>::dynarray(const dynarray<*T*>&)
lib.cons.dynarray.pt	17.5.5.1.5	dynarray<*T*>::dynarray(const *T**, size_t)
lib.cons.dynarray.t	17.5.5.1.4	dynarray<*T*>::dynarray(const *T*&, size_t)
lib.cons.filebuf	17.4.8.1.1	filebuf::filebuf()
lib.cons.float.complex.f.f	17.5.7.1.1.1	float_complex::float_complex(float, float)
lib.cons.ifstream	17.4.8.2.1	ifstream::ifstream()
lib.cons.ifstream.fn	17.4.8.2.2	ifstream::ifstream(const char*, openmode)
lib.cons.imanip.ios	17.4.5.2.1	imanip<*T*>::imanip(ios& (*)(ios&, *T*), *T*
lib.cons.invalidargument	17.3.2.5.1	invalidargument::invalidargument(const char*, const char*)
lib.cons.ios	17.4.1.1.45	ios::ios()
lib.cons.ios.sb	17.4.1.1.10	ios::ios(streambuf*)
lib.cons.ios::failure	17.4.1.1.1.1	ios::failure::failure(const char*, const char*)
lib.cons.ios::init	17.4.1.1.9.1	ios::Init::Init()
lib.cons.istdiostream.fi	17.4.8.5.1	istdiostream::istdiostream(*FILE**)
lib.cons.istream	17.4.3.1.1	istream::istream()
lib.cons.istringstream.m	17.4.7.2.1	istringstream::istringstream(ios::openmode)
lib.cons.istringstream.sm	17.4.7.2.2	istringstream::istringstream(const string&, ios::openmode
lib.cons.istrstream.cstr	17.4.6.2.1	istrstream::istrstream(const char*)
lib.cons.istrstream.cstrn	17.4.6.2.2	istrstream::istrstream(const char*, int)
lib.cons.istrstream.str	17.4.6.2.3	istrstream::istrstream(char*)
lib.cons.istrstream.strn	17.4.6.2.4	istrstream::istrstream(char*, int)
lib.cons.lengtherror	17.3.2.6.1	lengtherror::lengtherror(const char*, const char*)
lib.cons.long.double.complex.dc	17.5.7.3.1.3	long_double_complex::long_double_complex (double_complex&)
lib.cons.long.double.complex.fc	17.5.7.3.1.2	long_double_complex::long_double_complex (float_complex&)
lib.cons.long.double.complex.ld.ld	17.5.7.3.1.1	long_double_complex::long_double_complex(l double, long double)
lib.cons.ofstream	17.4.8.3.1	ofstream::ofstream()
lib.cons.ofstream.fn	17.4.8.3.2	ofstream::ofstream(const char*, openmode)
lib.cons.omanip.ios	17.4.5.3.1	omanip<*T*>::omanip(ios& (*)(ios&, *T*), *T*

lib.cons.ostdiostream.fi	17.4.8.6.1	ostdiostream::ostdiostream(*FILE**)
lib.cons.ostream.sb	17.4.4.1.1	ostream::ostream(streambuf*)
lib.cons.ostringstream.m	17.4.7.3.1	ostringstream::ostringstream(ios::openmode)
lib.cons.ostringstream.sm	17.4.7.3.2	ostringstream::ostringstream(const string&, ios::openmode
lib.cons.ostrstream	17.4.6.3.1	ostrstream::ostrstream()
lib.cons.ostrstream.str	17.4.6.3.2	ostrstream::ostrstream(char*, int, openmode)
lib.cons.outofrange	17.3.2.7.1	outofrange::outofrange(const char*, const char*)
lib.cons.overflow	17.3.2.8.1	overflow::overflow(const char*, const char*)
lib.cons.ptrdynarray	17.5.6.1.1	ptrdynarray<*T*>::ptrdynarray()
lib.cons.ptrdynarray.cap	17.5.6.1.2	ptrdynarray<*T*>::ptrdynarray(size_t, capacity)
lib.cons.ptrdynarray.pda	17.5.6.1.3	ptrdynarray<*T*>::ptrdynarray(const ptrdynarray<*T*>&)
lib.cons.ptrdynarray.ppt	17.5.6.1.5	ptrdynarray<*T*>::ptrdynarray(const *T***, size_t)
lib.cons.ptrdynarray.pt	17.5.6.1.4	ptrdynarray<*T*>::ptrdynarray(*T**)
lib.cons.smanip.ios	17.4.5.1.1	smanip<*T*>::smanip(ios& (*)(ios&, *T*), *T*
lib.cons.stdiobuf.fi	17.4.8.4.1	stdiobuf::stdiobuf(*FILE**)
lib.cons.streambuf	17.4.2.3.16	streambuf::streambuf()
lib.cons.streampos	17.4.2.2.1	streampos::streampos(streamoff)
lib.cons.string	17.5.1.1.1	string::string()
lib.cons.string.c	17.5.1.1.5	string::string(char, size_t)
lib.cons.string.cap	17.5.1.1.2	string::string(size_t, capacity)
lib.cons.string.sc	17.5.1.1.7	string(signed char, size_t)
lib.cons.string.str	17.5.1.1.4	string::string(const char*, size_t)
lib.cons.string.sub	17.5.1.1.3	string::string(const string&, size_t, size_t)
lib.cons.string.uc	17.5.1.1.6	string::string(unsigned char, size_t)
lib.cons.stringbuf.m	17.4.7.1.1	stringbuf::stringbuf(ios::openmode)
lib.cons.stringbuf.sm	17.4.7.1.2	stringbuf::stringbuf(const string&, ios::openmode)
lib.cons.strstreambuf.csstr	17.4.6.1.8	strstreambuf::strstreambuf(const signed char*, int)
lib.cons.strstreambuf.cstr	17.4.6.1.6	strstreambuf::strstreambuf(const char*, int)
lib.cons.strstreambuf.custr	17.4.6.1.7	strstreambuf::strstreambuf(const unsigned char*, int)
lib.cons.strstreambuf.ff	17.4.6.1.2	strstreambuf::strstreambuf(void* (*)(size_t), void (*)(void*))
lib.cons.strstreambuf.i	17.4.6.1.1	strstreambuf::strstreambuf(int)
lib.cons.strstreambuf.sstr	17.4.6.1.5	strstreambuf::strstreambuf(signed char*, int, signed char*)
lib.cons.strstreambuf.str	17.4.6.1.3	strstreambuf::strstreambuf(char*, int, char*)
lib.cons.strstreambuf.ustr	17.4.6.1.4	strstreambuf::strstreambuf(unsigned char*, int, unsigned char*)
lib.cons.typeinfo	17.3.4.2.6	typeinfo::typeinfo(const typeinfo&)
lib.cons.wstring	17.5.2.1.1	wstring::wstring()
lib.cons.wstring.cap	17.5.2.1.2	wstring::wstring(size_t, capacity)
lib.cons.wstring.wsub	17.5.2.1.3	wstring::wstring(const wstring&,

		`size_t, size_t)`
lib.cons.xalloc	17.3.3.1.1	`xalloc::xalloc(const char*, const char*)`
lib.cons.xdomain	17.3.2.9.1	`xdomain::xdomain(const char*, const char*, const char*)`
lib.cons.xlogic	17.3.2.2.1	`xlogic::xlogic(const char*, const char*, const char*)`
lib.cons.xmsg.sss	17.3.2.1.3	`xmsg::xmsg(const char*, const char*, const char*)`
lib.cons.xmsg.sssi	17.3.2.1.10	`xmsg::xmsg(const char*, const char*, const char*, int)`
lib.cons.xrange	17.3.2.10.1	`xrange::xrange(const char*, const char*, const char*)`
lib.cons.xruntime.sss	17.3.2.3.1	`xruntime::xruntime(const char*, const char*, const char*)`
lib.cons.xruntime.sssi	17.3.2.3.4	`xruntime::xruntime(const char*, const char*, const char*, int)`
lib.cos.dc	17.5.7.2.28	`cos(double_complex)`
lib.cos.fc	17.5.7.1.29	`cos(float_complex)`
lib.cos.ldc	17.5.7.3.27	`cos(long_double_complex)`
lib.cosh.dc	17.5.7.2.29	`cosh(double_complex)`
lib.cosh.fc	17.5.7.1.30	`cosh(float_complex)`
lib.cosh.ldc	17.5.7.3.28	`cosh(long_double_complex)`
lib.cout	17.4.9.2	Object `cout`
lib.dec	17.4.1.2	`dec(ios&)`
lib.definitions	17.1.5.12	Definitions
lib.derived.classes	17.1.5.10.3	Derived classes
lib.des.badcast	17.3.2.4.2	`badcast::~badcast()`
lib.des.badtypeid	17.3.4.1.2	`badtypeid::~badtypeid()`
lib.des.filebuf	17.4.8.1.2	`filebuf::~filebuf()`
lib.des.ifstream	17.4.8.2.3	`ifstream::~ifstream()`
lib.des.invalidargument	17.3.2.5.2	`invalidargument::~invalidargument()`
lib.des.ios	17.4.1.1.11	`ios::~ios()`
lib.des.ios::failure	17.4.1.1.1.2	`ios::failure::~failure()`
lib.des.ios::init	17.4.1.1.9.2	`ios::Init::~Init()`
lib.des.istdiostream	17.4.8.5.2	`istdiostream::~istdiostream()`
lib.des.istream	17.4.3.1.2	`istream::~istream()`
lib.des.istringstream	17.4.7.2.3	`istringstream::~istringstream()`
lib.des.istrstream	17.4.6.2.5	`istrstream::~istrstream()`
lib.des.lengtherror	17.3.2.6.2	`lengtherror::~lengtherror()`
lib.des.ofstream	17.4.8.3.3	`ofstream::~ofstream()`
lib.des.ostdiostream	17.4.8.6.2	`ostdiostream::~ostdiostream()`
lib.des.ostream	17.4.4.1.2	`ostream::~ostream()`
lib.des.ostringstream	17.4.7.3.3	`ostringstream::~ostringstream()`
lib.des.ostrstream	17.4.6.3.3	`ostrstream::~ostrstream()`
lib.des.outofrange	17.3.2.7.2	`outofrange::~outofrange()`
lib.des.overflow	17.3.2.8.2	`overflow::~overflow()`
lib.des.stdiobuf	17.4.8.4.2	`stdiobuf::~stdiobuf()`
lib.des.streambuf	17.4.2.3.1	`streambuf::~streambuf()`
lib.des.stringbuf	17.4.7.1.3	`stringbuf::~stringbuf()`
lib.des.strstreambuf	17.4.6.1.9	`strstreambuf::~strstreambuf()`
lib.des.typeinfo	17.3.4.2.1	`typeinfo::~typeinfo()`
lib.des.xalloc	17.3.3.1.2	`xalloc::~xalloc()`
lib.des.xdomain	17.3.2.9.2	`xdomain::~xdomain()`
lib.des.xlogic	17.3.2.2.2	`xlogic::~xlogic()`

lib.des.xmsg	17.3.2.1.4	xmsg::~xmsg()
lib.des.xrange	17.3.2.10.2	xrange::~xrange()
lib.des.xruntime	17.3.2.3.2	xruntime::~xruntime()
lib.double.complex	17.5.7.2.1	Class double_complex
lib.dynarray::append.pt	17.5.5.1.9	dynarray<*T*>::append(const *T**, size_t)
lib.dynarray::append.t	17.5.5.1.8	dynarray<*T*>::append(const *T*&, size_t)
lib.dynarray::assign.pt	17.5.5.1.11	dynarray<*T*>::assign(const *T**, size_t)
lib.dynarray::assign.t	17.5.5.1.10	dynarray<*T*>::assign(const *T*&, size_t)
lib.dynarray::base	17.5.5.1.21	dynarray<*T*>::base()
lib.dynarray::get.at	17.5.5.1.18	dynarray<*T*>::get_at(size_t)
lib.dynarray::insert.da	17.5.5.1.12	dynarray<*T*>::insert(size_t, const dynarray<*T*>&)
lib.dynarray::insert.pt	17.5.5.1.14	dynarray<*T*>::insert(size_t, const *T**, size_t)
lib.dynarray::insert.t	17.5.5.1.13	dynarray<*T*>::insert(size_t, const *T*&, size_t)
lib.dynarray::length	17.5.5.1.22	dynarray<*T*>::length()
lib.dynarray::op+=.da	17.5.5.1.6	dynarray<*T*>::operator+=(const dynarray<*T*>&)
lib.dynarray::op+=.t	17.5.5.1.7	dynarray<*T*>::operator+=(const *T*&)
lib.dynarray::op.array	17.5.5.1.20	dynarray<*T*>::operator[](size_t)
lib.dynarray::put.at	17.5.5.1.19	dynarray<*T*>::put_at(size_t, const *T*&)
lib.dynarray::remove	17.5.5.1.15	dynarray<*T*>::remove(size_t, size_t)
lib.dynarray::reserve	17.5.5.1.25	dynarray<*T*>::reserve()
lib.dynarray::reserve.cap	17.5.5.1.26	dynarray<*T*>::reserve(size_t)
lib.dynarray::resize	17.5.5.1.23	dynarray<*T*>::resize(size_t)
lib.dynarray::resize.t	17.5.5.1.24	dynarray<*T*>::resize(size_t, const *T*&)
lib.dynarray::sub.array	17.5.5.1.17	dynarray<*T*>::sub_array(dynarray<*T*>&, size_t, size_t)
lib.dynarray::swap	17.5.5.1.16	dynarray<*T*>::swap(dynarray<*T*>&)
lib.endl	17.4.4.2	endl(ostream&)
lib.ends	17.4.4.3	ends(ostream&)
lib.enumerated.types	17.1.5.10.1	Enumerated types
lib.exit	17.2.4.4	exit(int)
lib.exp.dc	17.5.7.2.30	exp(double_complex)
lib.exp.fc	17.5.7.1.31	exp(float_complex)
lib.exp.ldc	17.5.7.3.29	exp(long_double_complex)
lib.ext.bs	17.5.4.6	operator>>(istream&, bitstring&)
lib.ext.bt	17.5.3.5	operator>>(istream&, bits<*N*>&)
lib.ext.dc	17.5.7.2.23	operator>>(istream&, double_complex&)
lib.ext.fc	17.5.7.1.24	operator>>(istream&, float_complex&)
lib.ext.imanip	17.4.5.2.2	operator>>(istream&, const imanip<*T*>&)
lib.ext.ldc	17.5.7.3.22	operator>>(istream&, long_double_complex&)
lib.ext.smanip	17.4.5.1.2	operator>>(istream&, const smanip<*T*>&)
lib.ext.sub	17.5.1.17	operator>>(istream&, string&)
lib.filebuf	17.4.8.1	Class filebuf

lib.filebuf::close	17.4.8.1.6	`filebuf::close()`
lib.filebuf::is.open	17.4.8.1.3	`filebuf::is_open()`
lib.filebuf::open	17.4.8.1.4	`filebuf::open(const char*, ios::openmode)`
lib.filebuf::open.old	17.4.8.1.5	`filebuf::open(const char*, ios::open_mode)`
lib.filebuf::overflow	17.4.8.1.7	`filebuf::overflow(int)`
lib.filebuf::pbackfail	17.4.8.1.8	`filebuf::pbackfail(int)`
lib.filebuf::seekoff	17.4.8.1.13	`filebuf::seekoff(streamoff, ios::seekdir, ios::openmode)`
lib.filebuf::seekpos	17.4.8.1.14	`filebuf::seekpos(streampos, ios::openmode)`
lib.filebuf::setbuf	17.4.8.1.15	`filebuf::setbuf(char*, int)`
lib.filebuf::sync	17.4.8.1.16	`filebuf::sync()`
lib.filebuf::uflow	17.4.8.1.10	`filebuf::uflow()`
lib.filebuf::underflow	17.4.8.1.9	`filebuf::underflow()`
lib.filebuf::xsgetn	17.4.8.1.11	`filebuf::xsgetn(char*, int)`
lib.filebuf::xsputn	17.4.8.1.12	`filebuf::xsputn(const char*, int)`
lib.fixed	17.4.1.3	`fixed(ios&)`
lib.float.complex	17.5.7.1.1	Class `float_complex`
lib.flush	17.4.4.4	`flush(ostream&)`
lib.functions.within.classes	17.1.5.7	Functions within classes
lib.fvoid.t	17.3.1.1	Type `fvoid_t`
lib.getline.sub	17.5.1.18	`getline(istream&, string&, char)`
lib.global.functions	17.1.5.8	Global functions
lib.header.bits	17.5.3	Header `<bits>`
lib.header.bitstring	17.5.4	Header `<bitstring>`
lib.header.complex	17.5.7	Header `<complex>`
lib.header.defines	17.3.1	Header `<defines>`
lib.header.dynarray	17.5.5	Header `<dynarray>`
lib.header.exception	17.3.2	Header `<exception>`
lib.header.fstream	17.4.8	Header `<fstream>`
lib.header.iomanip	17.4.5	Header `<iomanip>`
lib.header.ios	17.4.1	Header `<ios>`
lib.header.iostream	17.4.9	Header `<iostream>`
lib.header.iso646.h	17.2.2.3	Header `<iso646.h>`
lib.header.istream	17.4.3	Header `<istream>`
lib.header.new	17.3.3	Header `<new>`
lib.header.ostream	17.4.4	Header `<ostream>`
lib.header.ptrdynarray	17.5.6	Header `<ptrdynarray>`
lib.header.sstream	17.4.7	Header `<sstream>`
lib.header.streambuf	17.4.2	Header `<streambuf>`
lib.header.string	17.5.1	Header `<string>`
lib.header.strstream	17.4.6	Header `<strstream>`
lib.header.typeinfo	17.3.4	Header `<typeinfo>`
lib.header.wstring	17.5.2	Header `<wstring>`
lib.headers	17.1.2	Headers
lib.hex	17.4.1.4	`hex(ios&)`
lib.ifstream	17.4.8.2	Class `ifstream`
lib.ifstream::close	17.4.8.2.8	`ifstream::close()`
lib.ifstream::is.open	17.4.8.2.5	`ifstream::is_open()`
lib.ifstream::open	17.4.8.2.6	`ifstream::open(const char*, openmode)`
lib.ifstream::open.old	17.4.8.2.7	`ifstream::open(const char*, open_mode)`

lib.ifstream::rdbuf	17.4.8.2.4	`ifstream::rdbuf()`
lib.imag.dc	17.5.7.2.31	`imag(double_complex)`
lib.imag.fc	17.5.7.1.32	`imag(float_complex)`
lib.imag.ldc	17.5.7.3.30	`imag(long_double_complex)`
lib.implementation.types	17.1.5.10	Implementation types
lib.input/output	17.4	Input/output
lib.ins.bs	17.5.4.7	`operator<<(ostream&, const bitstring&)`
lib.ins.bt	17.5.3.6	`operator<<(ostream&, const bits<N>&)`
lib.ins.dc	17.5.7.2.24	`operator<<(ostream&, double_complex)`
lib.ins.fc	17.5.7.1.25	`operator<<(ostream&, float_complex)`
lib.ins.ldc	17.5.7.3.23	`operator<<(ostream&, long_double_complex)`
lib.ins.omanip	17.4.5.3.2	`operator<<(istream&, const omanip<T>&)`
lib.ins.smanip	17.4.5.1.3	`operator<<(ostream&, const smanip<T>&)`
lib.ins.sub	17.5.1.19	`operator<<(ostream&, const string&)`
lib.instantiations.of.manipulators	17.4.5.4	Instantiations of manipulators
lib.internal	17.4.1.5	`internal(ios&)`
lib.intro.standard.c	17.1.1	Standard C library
lib.introduction	17.1	Introduction
lib.invalidargument	17.3.2.5	Class `invalidargument`
lib.invalidargument::do.raise	17.3.2.5.3	`invalidargument::do_raise()`
lib.ios	17.4.1.1	Class `ios`
lib.ios::bad	17.4.1.1.27	`ios::bad()`
lib.ios::clear.ios	17.4.1.1.20	`ios::clear(iostate)`
lib.ios::clear.ios.old	17.4.1.1.21	`ios::clear(io_state)`
lib.ios::copyfmt	17.4.1.1.14	`ios::copyfmt(const ios&)`
lib.ios::eof	17.4.1.1.25	`ios::eof()`
lib.ios::exceptions	17.4.1.1.28	`ios::exceptions()`
lib.ios::exceptions.ios	17.4.1.1.29	`ios::exceptions(iostate)`
lib.ios::exceptions.ios.old	17.4.1.1.30	`ios::exceptions(io_state)`
lib.ios::fail	17.4.1.1.26	`ios::fail()`
lib.ios::failure	17.4.1.1.1	Class `ios::failure`
lib.ios::failure::do.raise	17.4.1.1.1.3	`ios::failure::do_raise()`
lib.ios::fill	17.4.1.1.36	`ios::fill()`
lib.ios::fill.i	17.4.1.1.37	`ios::fill(int)`
lib.ios::flags	17.4.1.1.31	`ios::flags()`
lib.ios::flags.f	17.4.1.1.32	`ios::flags(fmtflags)`
lib.ios::fmtflags	17.4.1.1.2	Type `ios::fmtflags`
lib.ios::good	17.4.1.1.24	`ios::good()`
lib.ios::init	17.4.1.1.9	Class `ios::Init`
lib.ios::init.sb	17.4.1.1.46	`ios::init(streambuf*)`
lib.ios::io.state	17.4.1.1.6	Type `ios::io_state`
lib.ios::iostate	17.4.1.1.3	Type `ios::iostate`
lib.ios::iword	17.4.1.1.43	`ios::iword(int)`
lib.ios::open.mode	17.4.1.1.7	Type `ios::open_mode`
lib.ios::openmode	17.4.1.1.4	Type `ios::openmode`
lib.ios::operator!	17.4.1.1.13	`ios::operator!()`
lib.ios::operator.void*	17.4.1.1.12	`ios::operator void*()`
lib.ios::precision	17.4.1.1.38	`ios::precision()`
lib.ios::precision.i	17.4.1.1.39	`ios::precision(int)`
lib.ios::pword	17.4.1.1.44	`ios::pword(int)`

lib.ios::rdbuf	17.4.1.1.17	`ios::rdbuf()`
lib.ios::rdbuf.sb	17.4.1.1.18	`ios::rdbuf(streambuf*)`
lib.ios::rdstate	17.4.1.1.19	`ios::rdstate()`
lib.ios::seek.dir	17.4.1.1.8	Type `ios::seek_dir`
lib.ios::seekdir	17.4.1.1.5	Type `ios::seekdir`
lib.ios::setf.f	17.4.1.1.33	`ios::setf(fmtflags)`
lib.ios::setf.ff	17.4.1.1.34	`ios::setf(fmtflags, fmtflags)`
lib.ios::setstate.ios	17.4.1.1.22	`ios::setstate(iostate)`
lib.ios::setstate.ios.old	17.4.1.1.23	`ios::setstate(io_state)`
lib.ios::tie	17.4.1.1.15	`ios::tie()`
lib.ios::tie.os	17.4.1.1.16	`ios::tie(ostream*)`
lib.ios::unsetf	17.4.1.1.35	`ios::unsetf(fmtflags)`
lib.ios::width	17.4.1.1.40	`ios::width()`
lib.ios::width.i	17.4.1.1.41	`ios::width(int)`
lib.ios::xalloc	17.4.1.1.42	`ios::xalloc()`
lib.istdiostream	17.4.8.5	Class `istdiostream`
lib.istdiostream::buffered	17.4.8.5.4	`istdiostream::buffered()`
lib.istdiostream::buffered.i	17.4.8.5.5	`istdiostream::buffered(int)`
lib.istdiostream::rdbuf	17.4.8.5.3	`istdiostream::rdbuf()`
lib.istream	17.4.3.1	Class `istream`
lib.istream::ext.c	17.4.3.1.10	`istream::operator>>(char&)`
lib.istream::ext.d	17.4.3.1.20	`istream::operator>>(double&)`
lib.istream::ext.f	17.4.3.1.19	`istream::operator>>(float&)`
lib.istream::ext.i	17.4.3.1.15	`istream::operator>>(int&)`
lib.istream::ext.imanip	17.4.3.1.5	`istream::operator>>(istream& (*)(istream&))`
lib.istream::ext.iomanip	17.4.3.1.6	`istream::operator>>(ios& (*)(ios&))`
lib.istream::ext.ld	17.4.3.1.21	`istream::operator>>(long double&)`
lib.istream::ext.li	17.4.3.1.17	`istream::operator>>(long&)`
lib.istream::ext.ptr	17.4.3.1.22	`istream::operator>>(void*&)`
lib.istream::ext.sb	17.4.3.1.23	`istream::operator>>(streambuf&)`
lib.istream::ext.sc	17.4.3.1.12	`istream::operator>>(signed char&)`
lib.istream::ext.si	17.4.3.1.13	`istream::operator>>(short&)`
lib.istream::ext.sstr	17.4.3.1.9	`istream::operator>>(signed char*)`
lib.istream::ext.str	17.4.3.1.7	`istream::operator>>(char*)`
lib.istream::ext.uc	17.4.3.1.11	`istream::operator>>(unsigned char&)`
lib.istream::ext.ui	17.4.3.1.16	`istream::operator>>(unsigned int&)`
lib.istream::ext.uli	17.4.3.1.18	`istream::operator>>(unsigned long&)`
lib.istream::ext.usi	17.4.3.1.14	`istream::operator>>(unsigned short&)`
lib.istream::ext.ustr	17.4.3.1.8	`istream::operator>>(unsigned char*)`
lib.istream::gcount	17.4.3.1.42	`istream::gcount()`
lib.istream::get	17.4.3.1.24	`istream::get()`
lib.istream::get.c	17.4.3.1.28	`istream::get(char&)`
lib.istream::get.sb	17.4.3.1.31	`istream::get(streambuf&, char)`
lib.istream::get.sc	17.4.3.1.30	`istream::get(signed char&)`
lib.istream::get.sstr	17.4.3.1.27	`istream::get(signed char*, int, char)`
lib.istream::get.str	17.4.3.1.25	`istream::get(char*, int, char)`
lib.istream::get.uc	17.4.3.1.29	`istream::get(unsigned char&)`
lib.istream::get.ustr	17.4.3.1.26	`istream::get(unsigned char*, int, char)`
lib.istream::getline.sstr	17.4.3.1.34	`istream::getline(signed char*, int, char)`
lib.istream::getline.str	17.4.3.1.32	`istream::getline(char*, int, char)`
lib.istream::getline.ustr	17.4.3.1.33	`istream::getline(unsigned char*,`

		int, char)
lib.istream::ignore	17.4.3.1.35	istream::ignore(int, int)
lib.istream::ipfx	17.4.3.1.3	istream::ipfx(int)
lib.istream::isfx	17.4.3.1.4	istream::isfx()
lib.istream::peek	17.4.3.1.39	istream::peek()
lib.istream::putback	17.4.3.1.40	istream::putback(char)
lib.istream::read.sstr	17.4.3.1.38	istream::read(signed char*, int)
lib.istream::read.str	17.4.3.1.36	istream::read(char*, int)
lib.istream::read.ustr	17.4.3.1.37	istream::read(unsigned char*, int)
lib.istream::sync	17.4.3.1.43	istream::sync()
lib.istream::unget	17.4.3.1.41	istream::unget()
lib.istringstream	17.4.7.2	Class istringstream
lib.istringstream::rdbuf	17.4.7.2.4	istringstream::rdbuf()
lib.istringstream::str	17.4.7.2.5	istringstream::str()
lib.istringstream::str.s	17.4.7.2.6	istringstream::str(const string&)
lib.istrstream	17.4.6.2	Class istrstream
lib.istrstream::rdbuf	17.4.6.2.6	istrstream::rdbuf()
lib.language.support	17.3	Language support
lib.left	17.4.1.6	left(ios&)
lib.lengtherror	17.3.2.6	Class lengtherror
lib.lengtherror::do.raise	17.3.2.6.3	lengtherror::do_raise()
lib.library	17	Library
lib.log.dc	17.5.7.2.32	log(double_complex)
lib.log.fc	17.5.7.1.33	log(float_complex)
lib.log.ldc	17.5.7.3.31	log(long_double_complex)
lib.long.double.complex	17.5.7.3.1	Class long_double_complex
lib.longjmp	17.2.4.2	longjmp(jmp_buf, int)
lib.memchr	17.2.3.1	memchr(const void*, int, size_t)
lib.mods.to.behavior	17.2.4	Modifications to behavior
lib.mods.to.declarations	17.2.3	Modifications to declarations
lib.mods.to.definitions	17.2.2	Modifications to definitions
lib.mods.to.headers	17.2.1	Modifications to headers
lib.namespaces	17.1.3	Namespaces
lib.norm.dc	17.5.7.2.33	norm(double_complex)
lib.norm.fc	17.5.7.1.34	norm(float_complex)
lib.norm.ldc	17.5.7.3.32	norm(long_double_complex)
lib.noshowbase	17.4.1.7	noshowbase(ios&)
lib.noshowpoint	17.4.1.8	noshowpoint(ios&)
lib.noshowpos	17.4.1.9	noshowpos(ios&)
lib.noskipws	17.4.1.10	noskipws(ios&)
lib.nouppercase	17.4.1.11	nouppercase(ios&)
lib.null	17.2.2.2	Macro NULL
lib.objects.within.classes	17.1.5.5	Objects within classes
lib.oct	17.4.1.12	oct(ios&)
lib.offsetof	17.2.4.1	Macro offsetof
lib.ofstream	17.4.8.3	Class ofstream
lib.ofstream::close	17.4.8.3.8	ofstream::close()
lib.ofstream::is.open	17.4.8.3.5	ofstream::is_open()
lib.ofstream::open	17.4.8.3.6	ofstream::open(const char*, openmode)
lib.ofstream::open.old	17.4.8.3.7	ofstream::open(const char*, open_mode)
lib.ofstream::rdbuf	17.4.8.3.4	ofstream::rdbuf()
lib.op!=.c.sub	17.5.1.14	operator!=(char, const string&)
lib.op!=.d.dc	17.5.7.2.22	operator!=(double, double_complex)

lib.op!=.dc.d	17.5.7.2.21	operator!=(double_complex, double)
lib.op!=.dc.dc	17.5.7.2.20	operator!=(double_complex, double_complex)
lib.op!=.f.fc	17.5.7.1.23	operator!=(float, float_complex)
lib.op!=.fc.f	17.5.7.1.22	operator!=(float_complex, float)
lib.op!=.fc.fc	17.5.7.1.21	operator!=(float_complex, float_complex)
lib.op!=.ld.ldc	17.5.7.3.21	operator!=(long double, long_double_complex)
lib.op!=.ldc.ld	17.5.7.3.20	operator!=(long_double_complex, long double)
lib.op!=.ldc.ldc	17.5.7.3.19	operator!=(long_double_complex, long_double_complex)
lib.op!=.str.sub	17.5.1.13	operator!=(const char*, const string&)
lib.op!=.streampos	17.4.2.2.9	operator!=(streampos&)
lib.op!=.sub.c	17.5.1.16	operator!=(const string&, char)
lib.op!=.sub.str	17.5.1.15	operator!=(const string&, const char*)
lib.op!=.sub.sub	17.5.1.12	operator!=(const string&, const string&)
lib.op!=.wc.wsub	17.5.2.14	operator!=(wchar_t, const wstring&)
lib.op!=.wstr.wsub	17.5.2.13	operator!=(const wchar_t*, const wstring&)
lib.op!=.wsub.wc	17.5.2.16	operator!=(const wstring&, wchar_t)
lib.op!=.wsub.wstr	17.5.2.15	operator!=(const wstring&, const wchar_t*)
lib.op!=.wsub.wsub	17.5.2.12	operator!=(const wstring&, const wstring&)
lib.op&.bs.bs	17.5.4.3	operator&(const bitstring&, const bitstring&)
lib.op&.bt.bt	17.5.3.2	operator&(const bits<*N*>&, const bits<*N*>&)
lib.op*.d.dc	17.5.7.2.11	operator*(double, double_complex)
lib.op*.dc.d	17.5.7.2.10	operator*(double_complex, double)
lib.op*.dc.dc	17.5.7.2.9	operator*(double_complex, double_complex)
lib.op*.f.fc	17.5.7.1.12	operator*(float, float_complex)
lib.op*.fc.f	17.5.7.1.11	operator*(float_complex, float)
lib.op*.fc.fc	17.5.7.1.10	operator*(float_complex, float_complex)
lib.op*.ld.ldc	17.5.7.3.10	operator*(long double, long_double_complex)
lib.op*.ldc.ld	17.5.7.3.9	operator*(long_double_complex, long double)
lib.op*.ldc.ldc	17.5.7.3.8	operator*(long_double_complex, long_double_complex)
lib.op*=.dc	17.5.7.2.1.5	operator*=(double_complex)
lib.op*=.fc	17.5.7.1.1.4	operator*=(float_complex)
lib.op*=.ldc	17.5.7.3.1.6	operator*=(long_double_complex)
lib.op+.bs.bs	17.5.4.2	operator+(const bitstring&, const bitstring&)
lib.op+.c.sub	17.5.1.4	operator+(char, const string&)
lib.op+.d.dc	17.5.7.2.5	operator+(double, double_complex)
lib.op+.da.da	17.5.5.2	operator+(const dynarray<*T*>&,

		const dynarray<T>&)
lib.op+.da.t	17.5.5.3	operator+(const dynarray<T>&, const T&)
lib.op+.dc	17.5.7.2.15	operator+(double_complex)
lib.op+.dc.d	17.5.7.2.4	operator+(double_complex, double)
lib.op+.dc.dc	17.5.7.2.3	operator+(double_complex, double_complex)
lib.op+.f.fc	17.5.7.1.6	operator+(float, float_complex)
lib.op+.fc	17.5.7.1.16	operator+(float_complex)
lib.op+.fc.f	17.5.7.1.5	operator+(float_complex, float)
lib.op+.fc.fc	17.5.7.1.4	operator+(float_complex, float_complex)
lib.op+.ld.ldc	17.5.7.3.4	operator+(long double, long_double_complex)
lib.op+.ldc	17.5.7.3.14	operator+(long_double_complex)
lib.op+.ldc.ld	17.5.7.3.3	operator+(long_double_complex, long double)
lib.op+.ldc.ldc	17.5.7.3.2	operator+(long_double_complex, long_double_complex)
lib.op+.pda.pda	17.5.6.2	operator+(const ptrdynarray<T>&, const ptrdynarray<T>&)
lib.op+.pda.pt	17.5.6.3	operator+(const ptrdynarray<T>&, T*)
lib.op+.pt.pda	17.5.6.4	operator+(T*, const ptrdynarray<T>&)
lib.op+.str.c	17.5.1.6	operator+(const string&, char)
lib.op+.str.sub	17.5.1.3	operator+(const char*, const string&)
lib.op+.sub.str	17.5.1.5	operator+(const string&, const char*)
lib.op+.sub.sub	17.5.1.2	operator+(const string&, const string&)
lib.op+.t.da	17.5.5.4	operator+(const T&, const dynarray<T>&)
lib.op+.wc.wsub	17.5.2.4	operator+(wchar_t, const wstring&)
lib.op+.wstr.wsub	17.5.2.3	operator+(const wchar_t*, const wstring&)
lib.op+.wsub.wc	17.5.2.6	operator+(const wstring&, wchar_t)
lib.op+.wsub.wstr	17.5.2.5	operator+(const wstring&, const wchar_t*)
lib.op+.wsub.wsub	17.5.2.2	operator+(const wstring&, const wstring&)
lib.op+=.dc	17.5.7.2.1.3	operator+=(double_complex)
lib.op+=.fc	17.5.7.1.1.2	operator+=(float_complex)
lib.op+=.ldc	17.5.7.3.1.4	operator+=(long_double_complex)
lib.op-.d.dc	17.5.7.2.8	operator-(double, double_complex)
lib.op-.dc	17.5.7.2.16	operator-(double_complex)
lib.op-.dc.d	17.5.7.2.7	operator-(double_complex, double)
lib.op-.dc.dc	17.5.7.2.6	operator-(double_complex, double_complex)
lib.op-.f.fc	17.5.7.1.9	operator-(float, float_complex)
lib.op-.fc	17.5.7.1.17	operator-(float_complex)
lib.op-.fc.f	17.5.7.1.8	operator-(float_complex, float)
lib.op-.fc.fc	17.5.7.1.7	operator-(float_complex, float_complex)
lib.op-.ld.ldc	17.5.7.3.7	operator-(long double, long_double_complex)

lib.op-.ldc	17.5.7.3.15	`operator-(long_double_complex)`
lib.op-.ldc.ld	17.5.7.3.6	`operator-(long_double_complex, long double)`
lib.op-.ldc.ldc	17.5.7.3.5	`operator-(long_double_complex, long_double_complex)`
lib.op-=.dc	17.5.7.2.1.4	`operator-=(double_complex)`
lib.op-=.fc	17.5.7.1.1.3	`operator-=(float_complex)`
lib.op-=.ldc	17.5.7.3.1.5	`operator-=(long_double_complex)`
lib.op.delete	17.3.3.3	`operator delete(void*)`
lib.op.delete.array	17.3.3.4	`operator delete[](void*)`
lib.op.new	17.3.3.5	`operator new(size_t)`
lib.op.new.array	17.3.3.6	`operator new[](size_t)`
lib.op/.d.dc	17.5.7.2.14	`operator/(double, double_complex)`
lib.op/.dc.d	17.5.7.2.13	`operator/(double_complex, double)`
lib.op/.dc.dc	17.5.7.2.12	`operator/(double_complex, double_complex)`
lib.op/.f.fc	17.5.7.1.15	`operator/(float, float_complex)`
lib.op/.fc.f	17.5.7.1.14	`operator/(float_complex, float)`
lib.op/.fc.fc	17.5.7.1.13	`operator/(float_complex, float_complex)`
lib.op/.ld.ldc	17.5.7.3.13	`operator/(long double, long_double_complex)`
lib.op/.ldc.ld	17.5.7.3.12	`operator/(long_double_complex, long double)`
lib.op/.ldc.ldc	17.5.7.3.11	`operator/(long_double_complex, long_double_complex)`
lib.op/=.dc	17.5.7.2.1.6	`operator/=(double_complex)`
lib.op/=.fc	17.5.7.1.1.5	`operator/=(float_complex)`
lib.op/=.ldc	17.5.7.3.1.7	`operator/=(long_double_complex)`
lib.op==.c.sub	17.5.1.9	`operator==(char, const string&)`
lib.op==.d.dc	17.5.7.2.19	`operator==(double, double_complex)`
lib.op==.dc.d	17.5.7.2.18	`operator==(double_complex, double)`
lib.op==.dc.dc	17.5.7.2.17	`operator==(double_complex, double_complex)`
lib.op==.f.fc	17.5.7.1.20	`operator==(float, float_complex)`
lib.op==.fc.f	17.5.7.1.19	`operator==(float_complex, float)`
lib.op==.fc.fc	17.5.7.1.18	`operator==(float_complex, float_complex)`
lib.op==.ld.ldc	17.5.7.3.18	`operator==(long double, long_double_complex)`
lib.op==.ldc.ld	17.5.7.3.17	`operator==(long_double_complex, long double)`
lib.op==.ldc.ldc	17.5.7.3.16	`operator==(long_double_complex, long_double_complex)`
lib.op==.str.sub	17.5.1.8	`operator==(const char*, const string&)`
lib.op==.sub.c	17.5.1.11	`operator==(const string&, char)`
lib.op==.sub.str	17.5.1.10	`operator==(const string&, const char*)`
lib.op==.sub.sub	17.5.1.7	`operator==(const string&, const string&)`
lib.op==.wc.wsub	17.5.2.9	`operator==(wchar_t, const wstring&)`
lib.op==.wstr.wsub	17.5.2.8	`operator==(const wchar_t*, const wstring&)`
lib.op==.wsub.wc	17.5.2.11	`operator==(const wstring&, wchar_t)`

lib.op==.wsub.wstr	17.5.2.10	operator==(const wstring&, const wchar_t*)
lib.op==.wsub.wsub	17.5.2.7	operator==(const wstring&, const wstring&)
lib.op^.bs.bs	17.5.4.5	operator^(const bitstring&, const bitstring&)
lib.op^.bt.bt	17.5.3.4	operator^(const bits<N>&, const bits<N>&)
lib.optional.members	17.1.5.6	Optional members
lib.op|.bs.bs	17.5.4.4	operator|(const bitstring&, const bitstring&)
lib.op|.bt.bt	17.5.3.3	operator|(const bits<N>&, const bits<N>&)
lib.ostdiostream	17.4.8.6	Class ostdiostream
lib.ostdiostream::buffered	17.4.8.6.4	ostdiostream::buffered()
lib.ostdiostream::buffered.i	17.4.8.6.5	ostdiostream::buffered(int)
lib.ostdiostream::rdbuf	17.4.8.6.3	ostdiostream::rdbuf()
lib.ostream	17.4.4.1	Class ostream
lib.ostream::flush	17.4.4.1.26	ostream::flush()
lib.ostream::ins.c	17.4.4.1.8	ostream::operator<<(char)
lib.ostream::ins.d	17.4.4.1.18	ostream::operator<<(double)
lib.ostream::ins.f	17.4.4.1.17	ostream::operator<<(float)
lib.ostream::ins.i	17.4.4.1.13	ostream::operator<<(int)
lib.ostream::ins.iomanip	17.4.4.1.6	ostream::operator<<(ios& (*)(ios&))
lib.ostream::ins.ld	17.4.4.1.19	ostream::operator<<(long double)
lib.ostream::ins.li	17.4.4.1.15	ostream::operator<<(long)
lib.ostream::ins.omanip	17.4.4.1.5	ostream::operator<<(ostream& (*)(ostream&))
lib.ostream::ins.ptr	17.4.4.1.20	ostream::operator<<(void*)
lib.ostream::ins.sb	17.4.4.1.21	ostream::operator<<(streambuf&)
lib.ostream::ins.sc	17.4.4.1.10	ostream::operator<<(signed char)
lib.ostream::ins.si	17.4.4.1.11	ostream::operator<<(short)
lib.ostream::ins.str	17.4.4.1.7	ostream::operator<<(const char*)
lib.ostream::ins.uc	17.4.4.1.9	ostream::operator<<(unsigned char)
lib.ostream::ins.ui	17.4.4.1.14	ostream::operator<<(unsigned int)
lib.ostream::ins.uli	17.4.4.1.16	ostream::operator<<(unsigned long)
lib.ostream::ins.usi	17.4.4.1.12	ostream::operator<<(unsigned short)
lib.ostream::opfx	17.4.4.1.3	ostream::opfx()
lib.ostream::osfx	17.4.4.1.4	ostream::osfx()
lib.ostream::put	17.4.4.1.22	ostream::put(char)
lib.ostream::write.sstr	17.4.4.1.25	ostream::write(const signed char*, int)
lib.ostream::write.str	17.4.4.1.23	ostream::write(const char*, int)
lib.ostream::write.ustr	17.4.4.1.24	ostream::write(const unsigned char*, int)
lib.ostringstream	17.4.7.3	Class ostringstream
lib.ostringstream::rdbuf	17.4.7.3.4	ostringstream::rdbuf()
lib.ostringstream::str	17.4.7.3.5	ostringstream::str()
lib.ostringstream::str.s	17.4.7.3.6	ostringstream::str(const string&)
lib.ostrstream	17.4.6.3	Class ostrstream
lib.ostrstream::freeze	17.4.6.3.5	ostrstream::freeze(int)
lib.ostrstream::pcount	17.4.6.3.7	ostrstream::pcount()
lib.ostrstream::rdbuf	17.4.6.3.4	ostrstream::rdbuf()
lib.ostrstream::str	17.4.6.3.6	ostrstream::str()
lib.outofrange	17.3.2.7	Class outofrange

lib.outofrange::do.raise	17.3.2.7.3	outofrange::do_raise()
lib.overflow	17.3.2.8	Class overflow
lib.overflow::do.raise	17.3.2.8.3	overflow::do_raise()
lib.placement.op.new	17.3.3.7	operator new(size_t, void*)
lib.placement.op.new.array	17.3.3.8	operator new[](size_t, void*)
lib.polar.d.d	17.5.7.2.34	polar(double, double)
lib.polar.f.f	17.5.7.1.35	polar(float, float)
lib.polar.ld.ld	17.5.7.3.33	polar(long double, long double)
lib.pow.d.dc	17.5.7.2.38	pow(double, double_complex)
lib.pow.dc.d	17.5.7.2.36	pow(double_complex, double)
lib.pow.dc.dc	17.5.7.2.35	pow(double_complex, double_complex)
lib.pow.dc.i	17.5.7.2.37	pow(double_complex, int)
lib.pow.f.fc	17.5.7.1.39	pow(float, float_complex)
lib.pow.fc.f	17.5.7.1.37	pow(float_complex, float)
lib.pow.fc.fc	17.5.7.1.36	pow(float_complex, float_complex)
lib.pow.fc.i	17.5.7.1.38	pow(float_complex, int)
lib.pow.ld.ldc	17.5.7.3.37	pow(long double, long_double_complex)
lib.pow.ldc.i	17.5.7.3.36	pow(long_double_complex, int)
lib.pow.ldc.ld	17.5.7.3.35	pow(long_double_complex, long double)
lib.pow.ldc.ldc	17.5.7.3.34	pow(long_double_complex, long_double_complex)
lib.protection.within.classes	17.1.5.11	Protection within classes
lib.ptrdiff.t	17.3.1.2	Type ptrdiff_t
lib.ptrdynarray::append.ppt	17.5.6.1.9	ptrdynarray<*T*>::append(*T***, size_t)
lib.ptrdynarray::append.pt	17.5.6.1.8	ptrdynarray<*T*>::append(*T**, size_t)
lib.ptrdynarray::assign.ppt	17.5.6.1.11	ptrdynarray<*T*>::assign(*T***, size_t)
lib.ptrdynarray::assign.pt	17.5.6.1.10	ptrdynarray<*T*>::assign(*T**, size_t)
lib.ptrdynarray::base	17.5.6.1.21	ptrdynarray<*T*>::base()
lib.ptrdynarray::get.at	17.5.6.1.18	ptrdynarray<*T*>::get_at(size_t)
lib.ptrdynarray::insert.pda	17.5.6.1.12	ptrdynarray<*T*>::insert(size_t, const ptrdynarray<*T*>&, size_t)
lib.ptrdynarray::insert.ppt	17.5.6.1.14	ptrdynarray<*T*>::insert(size_t, *T***, size_t)
lib.ptrdynarray::insert.pt	17.5.6.1.13	ptrdynarray<*T*>::insert(size_t, *T**, size_t)
lib.ptrdynarray::length	17.5.6.1.22	ptrdynarray<*T*>::length()
lib.ptrdynarray::op+=.pda	17.5.6.1.6	ptrdynarray<*T*>::operator+=(const ptrdynarray<*T*>&)
lib.ptrdynarray::op+=.pt	17.5.6.1.7	ptrdynarray<*T*>::operator+=(*T**)
lib.ptrdynarray::op.array	17.5.6.1.20	ptrdynarray<*T*>::operator[](size_t)
lib.ptrdynarray::put.at	17.5.6.1.19	ptrdynarray<*T*>::put_at(size_t, const *T*&)
lib.ptrdynarray::remove	17.5.6.1.15	ptrdynarray<*T*>::remove(size_t, size_t)
lib.ptrdynarray::reserve	17.5.6.1.25	ptrdynarray<*T*>::reserve()
lib.ptrdynarray::reserve.cap	17.5.6.1.26	ptrdynarray<*T*>::reserve(size_t)
lib.ptrdynarray::resize	17.5.6.1.23	ptrdynarray<*T*>::resize(size_t)
lib.ptrdynarray::resize.pt	17.5.6.1.24	ptrdynarray<*T*>::resize(size_t, *T**)
lib.ptrdynarray::sub.array	17.5.6.1.17	ptrdynarray<*T*>::sub_array(ptrdynarray<*T*>&, size_t, size_t)
lib.ptrdynarray::swap	17.5.6.1.16	ptrdynarray<*T*>::swap(ptrdynarray<*T*>&)
lib.real.dc	17.5.7.2.39	real(double_complex)
lib.real.fc	17.5.7.1.40	real(float_complex)
lib.real.ldc	17.5.7.3.38	real(long_double_complex)

lib.res.and.conventions	17.1.5	Restrictions and conventions
lib.res.on.arguments	17.1.5.2	Restrictions on arguments
lib.res.on.exception.handling	17.1.5.3	Restrictions on exception handling
lib.res.on.macro.definitions	17.1.5.1	Restrictions on macro definitions
lib.reserved.names	17.1.4	Reserved names
lib.resetiosflags	17.4.5.4.1	`resetiosflags(ios::fmtflags)`
lib.right	17.4.1.13	`right(ios&)`
lib.scientific	17.4.1.14	`scientific(ios&)`
lib.set.new.handler	17.3.3.2	`set_new_handler(fvoid_t*)`
lib.set.terminate	17.3.2.11	`set_terminate(fvoid_t*)`
lib.set.unexpected	17.3.2.12	`set_unexpected(fvoid_t*)`
lib.setbase	17.4.5.4.3	`setbase(int)`
lib.setfill	17.4.5.4.4	`setfill(int)`
lib.setiosflags	17.4.5.4.2	`setiosflags(ios::fmtflags)`
lib.setprecision	17.4.5.4.5	`setprecision(int)`
lib.setw	17.4.5.4.6	`setw(int)`
lib.showbase	17.4.1.15	`showbase(ios&)`
lib.showpoint	17.4.1.16	`showpoint(ios&)`
lib.showpos	17.4.1.17	`showpos(ios&)`
lib.sin.dc	17.5.7.2.40	`sin(double_complex)`
lib.sin.fc	17.5.7.1.41	`sin(float_complex)`
lib.sin.ldc	17.5.7.3.39	`sin(long_double_complex)`
lib.sinh.dc	17.5.7.2.41	`sinh(double_complex)`
lib.sinh.fc	17.5.7.1.42	`sinh(float_complex)`
lib.sinh.ldc	17.5.7.3.40	`sinh(long_double_complex)`
lib.size.t	17.3.1.3	Type `size_t`
lib.skipws	17.4.1.18	`skipws(ios&)`
lib.sqrt.dc	17.5.7.2.42	`sqrt(double_complex)`
lib.sqrt.fc	17.5.7.1.43	`sqrt(float_complex)`
lib.sqrt.ldc	17.5.7.3.41	`sqrt(long_double_complex)`
lib.standard.c.library	17.2	Standard C library
lib.stdiobuf	17.4.8.4	Class `stdiobuf`
lib.stdiobuf::buffered	17.4.8.4.3	`stdiobuf::buffered()`
lib.stdiobuf::buffered.i	17.4.8.4.4	`stdiobuf::buffered(int)`
lib.stdiobuf::overflow	17.4.8.4.5	`stdiobuf::overflow(int)`
lib.stdiobuf::pbackfail	17.4.8.4.6	`stdiobuf::pbackfail(int)`
lib.stdiobuf::seekoff	17.4.8.4.11	`stdiobuf::seekoff(streamoff, ios::seekdir, ios::openmode)`
lib.stdiobuf::seekpos	17.4.8.4.12	`stdiobuf::seekpos(streampos, ios::openmode)`
lib.stdiobuf::setbuf	17.4.8.4.13	`stdiobuf::setbuf(char*, int)`
lib.stdiobuf::sync	17.4.8.4.14	`stdiobuf::sync()`
lib.stdiobuf::uflow	17.4.8.4.8	`stdiobuf::uflow()`
lib.stdiobuf::underflow	17.4.8.4.7	`stdiobuf::underflow()`
lib.stdiobuf::xsgetn	17.4.8.4.9	`stdiobuf::xsgetn(char*, int)`
lib.stdiobuf::xsputn	17.4.8.4.10	`stdiobuf::xsputn(const char*, int)`
lib.storage.allocation.functions	17.2.4.3	Storage allocation functions
lib.strchr	17.2.3.2	`strchr(const char*, int)`
lib.streambuf	17.4.2.3	Class `streambuf`
lib.streambuf::eback	17.4.2.3.17	`streambuf::eback()`
lib.streambuf::egptr	17.4.2.3.19	`streambuf::egptr()`
lib.streambuf::epptr	17.4.2.3.24	`streambuf::epptr()`
lib.streambuf::gbump	17.4.2.3.20	`streambuf::gbump(int)`
lib.streambuf::gptr	17.4.2.3.18	`streambuf::gptr()`
lib.streambuf::overflow	17.4.2.3.27	`streambuf::overflow(int)`

lib.streambuf::pbackfail	17.4.2.3.28	`streambuf::pbackfail(int)`
lib.streambuf::pbase	17.4.2.3.22	`streambuf::pbase()`
lib.streambuf::pbump	17.4.2.3.25	`streambuf::pbump(int)`
lib.streambuf::pptr	17.4.2.3.23	`streambuf::pptr()`
lib.streambuf::pubseekoff	17.4.2.3.2	`streambuf::pubseekoff(streamoff, ios::seekdir, ios::openmode)`
lib.streambuf::pubseekoff.old	17.4.2.3.3	`streambuf::pubseekoff(streamoff, ios::seek_dir, ios::open_mode)`
lib.streambuf::pubseekpos	17.4.2.3.4	`streambuf::pubseekpos(streampos, ios::openmode)`
lib.streambuf::pubseekpos.old	17.4.2.3.5	`streambuf::pubseekpos(streampos, ios::open_mode)`
lib.streambuf::pubsetbuf	17.4.2.3.6	`streambuf::pubsetbuf(char*, int)`
lib.streambuf::pubsync	17.4.2.3.7	`streambuf::pubsync()`
lib.streambuf::sbumpc	17.4.2.3.8	`streambuf::sbumpc()`
lib.streambuf::seekoff	17.4.2.3.33	`streambuf::seekoff(streamoff, ios::seekdir, ios::openmode)`
lib.streambuf::seekpos	17.4.2.3.34	`streambuf::seekpos(streampos, ios::openmode)`
lib.streambuf::setbuf	17.4.2.3.35	`streambuf::setbuf(char*, int)`
lib.streambuf::setg	17.4.2.3.21	`streambuf::setg(char*, char*, char*)`
lib.streambuf::setp	17.4.2.3.26	`streambuf::setp(char*, char*)`
lib.streambuf::sgetc	17.4.2.3.9	`streambuf::sgetc()`
lib.streambuf::sgetn	17.4.2.3.10	`streambuf::sgetn(char*, int)`
lib.streambuf::snextc	17.4.2.3.11	`streambuf::snextc()`
lib.streambuf::sputbackc	17.4.2.3.12	`streambuf::sputbackc(char)`
lib.streambuf::sputc	17.4.2.3.14	`streambuf::sputc(int)`
lib.streambuf::sputn	17.4.2.3.15	`streambuf::sputn(const char*, int)`
lib.streambuf::sungetc	17.4.2.3.13	`streambuf::sungetc()`
lib.streambuf::sync	17.4.2.3.36	`streambuf::sync()`
lib.streambuf::uflow	17.4.2.3.30	`streambuf::uflow()`
lib.streambuf::underflow	17.4.2.3.29	`streambuf::underflow()`
lib.streambuf::xsgetn	17.4.2.3.31	`streambuf::xsgetn(char*, int)`
lib.streambuf::xsputn	17.4.2.3.32	`streambuf::xsputn(const char*, int)`
lib.streamoff	17.4.2.1	Type `streamoff`
lib.streamos::op-=	17.4.2.2.5	`streampos::operator-=(streamoff)`
lib.streampos	17.4.2.2	Class `streampos`
lib.streampos::offset	17.4.2.2.2	`streampos::offset()`
lib.streampos::op+	17.4.2.2.6	`streampos::operator+(streamoff)`
lib.streampos::op+=	17.4.2.2.4	`streampos::operator+=(streamoff)`
lib.streampos::op-.off	17.4.2.2.7	`streampos::operator-(streamoff)`
lib.streampos::op-.sp	17.4.2.2.3	`streampos::operator-(streampos&)`
lib.streampos::op==	17.4.2.2.8	`streampos::operator==(streampos&)`
lib.string	17.5.1.1	Class `string`
lib.string::append.c	17.5.1.1.15	`string::append(char, size_t)`
lib.string::append.str	17.5.1.1.14	`string::append(const char*, size_t)`
lib.string::append.sub	17.5.1.1.13	`string::append(const string&, size_t, size_t)`
lib.string::assign.c	17.5.1.1.18	`string::assign(char, size_t)`
lib.string::assign.str	17.5.1.1.17	`string::assign(const char*, size_t)`
lib.string::assign.sub	17.5.1.1.16	`string::assign(const string&, size_t, size_t)`
lib.string::c.str	17.5.1.1.29	`string::c_str()`
lib.string::compare.c	17.5.1.1.56	`string::compare(char, size_t)`
lib.string::compare.str	17.5.1.1.55	`string::compare(const char*, size_t)`

lib.string::compare.sub	17.5.1.1.54	`string::compare(const string&, size_t, size_t)`
lib.string::copy	17.5.1.1.34	`string::copy(char*, size_t, size_t)`
lib.string::find.c	17.5.1.1.37	`string::find(char, size_t)`
lib.string::find.first.not.of.c	17.5.1.1.49	`string::find_first_not_of(char, size_t)`
lib.string::find.first.not.of.str	17.5.1.1.48	`string::find_first_not_of(const char*, size_t,size_t)`
lib.string::find.first.not.of.sub	17.5.1.1.47	`string::find_first_not_of(const string&,size_t)`
lib.string::find.first.of.c	17.5.1.1.43	`string::find_first_of(char, size_t)`
lib.string::find.first.of.str	17.5.1.1.42	`string::find_first_of(const char*, size_t,size_t)`
lib.string::find.first.of.sub	17.5.1.1.41	`string::find_first_of(const string&, size_t)`
lib.string::find.last.not.of.c	17.5.1.1.52	`string::find_last_not_of(char, size_t)`
lib.string::find.last.not.of.str	17.5.1.1.51	`string::find_last_not_of(const char*, size_t, size_t)`
lib.string::find.last.not.of.sub	17.5.1.1.50	`string::find_last_not_of(const string&, size_t)`
lib.string::find.last.of.c	17.5.1.1.46	`string::find_last_of(char, size_t)`
lib.string::find.last.of.str	17.5.1.1.45	`string::find_last_of(const char*, size_t,size_t)`
lib.string::find.last.of.sub	17.5.1.1.44	`string::find_last_of(const string&, size_t)`
lib.string::find.str	17.5.1.1.36	`string::find(const char*, size_t, size_t)`
lib.string::find.sub	17.5.1.1.35	`string::find(const string&, size_t)`
lib.string::get.at	17.5.1.1.26	`string::get_at(size_t)`
lib.string::insert.c	17.5.1.1.21	`string::insert(size_t, char, size_t)`
lib.string::insert.str	17.5.1.1.20	`string::insert(size_t, const char*, size_t)`
lib.string::insert.sub	17.5.1.1.19	`string::insert(size_t, const string&, size_t, size_t)`
lib.string::length	17.5.1.1.30	`string::length()`
lib.string::op+=.c	17.5.1.1.12	`string::operator+=(char)`
lib.string::op+=.str	17.5.1.1.11	`string::operator+=(const char*)`
lib.string::op+=.sub	17.5.1.1.10	`string::operator+=(const string&)`
lib.string::op.array	17.5.1.1.28	`string::operator[](size_t)`
lib.string::op=.c	17.5.1.1.9	`string::operator=(char)`
lib.string::op=.str	17.5.1.1.8	`string::operator=(const char*)`
lib.string::put.at	17.5.1.1.27	`string::put_at(size_t, char)`
lib.string::remove	17.5.1.1.22	`string::remove(size_t, size_t)`
lib.string::replace.c	17.5.1.1.25	`string::replace(size_t, size_t, char, size_t)`
lib.string::replace.str	17.5.1.1.24	`string::replace(size_t, size_t, const char*, size_t)`
lib.string::replace.sub	17.5.1.1.23	`string::replace(size_t, size_t, const string&, size_t, size_t)`
lib.string::reserve	17.5.1.1.32	`string::reserve()`
lib.string::reserve.cap	17.5.1.1.33	`string::reserve(size_t)`
lib.string::resize	17.5.1.1.31	`string::resize(size_t, char)`
lib.string::rfind.c	17.5.1.1.40	`string::rfind(char, size_t)`
lib.string::rfind.str	17.5.1.1.39	`string::rfind(const char*, size_t, size_t)`

lib.string::rfind.sub	17.5.1.1.38	string::rfind(const string&, size_t)
lib.string::substr	17.5.1.1.53	string::substr(size_t, size_t)
lib.stringbuf	17.4.7.1	Class stringbuf
lib.stringbuf::overflow	17.4.7.1.6	stringbuf::overflow(int)
lib.stringbuf::pbackfail	17.4.7.1.7	stringbuf::pbackfail(int)
lib.stringbuf::seekoff	17.4.7.1.12	stringbuf::seekoff(streamoff, ios::seekdir, ios::openmode)
lib.stringbuf::seekpos	17.4.7.1.13	stringbuf::seekpos(streampos, ios::openmode)
lib.stringbuf::setbuf	17.4.7.1.14	stringbuf::setbuf(char*, int)
lib.stringbuf::str	17.4.7.1.4	stringbuf::str()
lib.stringbuf::str.s	17.4.7.1.5	stringbuf::str(const string&)
lib.stringbuf::sync	17.4.7.1.15	stringbuf::sync()
lib.stringbuf::uflow	17.4.7.1.9	stringbuf::uflow()
lib.stringbuf::underflow	17.4.7.1.8	stringbuf::underflow()
lib.stringbuf::xsgetn	17.4.7.1.10	stringbuf::xsgetn(char*, int)
lib.stringbuf::xsputn	17.4.7.1.11	stringbuf::xsputn(const char*, int)
lib.strpbrk	17.2.3.3	strpbrk(const char*, const char*)
lib.strrchr	17.2.3.4	strrchr(const char*, int)
lib.strstr	17.2.3.5	strstr(const char*, const char*)
lib.strstreambuf	17.4.6.1	Class strstreambuf
lib.strstreambuf::freeze	17.4.6.1.10	strstreambuf::freeze(int)
lib.strstreambuf::overflow	17.4.6.1.13	strstreambuf::overflow(int)
lib.strstreambuf::pbackfail	17.4.6.1.14	strstreambuf::pbackfail(int)
lib.strstreambuf::pcount	17.4.6.1.12	strstreambuf::pcount()
lib.strstreambuf::seekoff	17.4.6.1.19	strstreambuf::seekoff(streamoff, ios::seekdir, ios::openmode)
lib.strstreambuf::seekpos	17.4.6.1.20	strstreambuf::seekpos(streampos, ios::openmode)
lib.strstreambuf::setbuf	17.4.6.1.21	strstreambuf::setbuf(char*, int)
lib.strstreambuf::str	17.4.6.1.11	strstreambuf::str()
lib.strstreambuf::sync	17.4.6.1.22	strstreambuf::sync()
lib.strstreambuf::uflow	17.4.6.1.16	strstreambuf::uflow()
lib.strstreambuf::underflow	17.4.6.1.15	strstreambuf::underflow()
lib.strstreambuf::xsgetn	17.4.6.1.17	strstreambuf::xsgetn(char*, int)
lib.strstreambuf::xsputn	17.4.6.1.18	strstreambuf::xsputn(const char*, int)
lib.support.classes	17.5	Support classes
lib.template.bits	17.5.3.1	Template class bits<*N*>
lib.template.dynarray	17.5.5.1	Template class dynarray<*T*>
lib.template.imanip	17.4.5.2	Template class imanip<*T*>
lib.template.omanip	17.4.5.3	Template class omanip<*T*>
lib.template.ptrdynarray	17.5.6.1	Template class ptrdynarray<*T*>
lib.template.smanip	17.4.5.1	Template class smanip<*T*>
lib.terminate	17.3.2.13	terminate()
lib.typeinfo	17.3.4.2	Class typeinfo
lib.typeinfo::before	17.3.4.2.4	typeinfo::before(const typeinfo&)
lib.typeinfo::name	17.3.4.2.5	typeinfo::name()
lib.typeinfo::op!=	17.3.4.2.3	typeinfo::operator!=(const typeinfo&)
lib.typeinfo::op=	17.3.4.2.7	typeinfo::operator=(const typeinfo&)
lib.typeinfo::op==	17.3.4.2.2	typeinfo::operator==(const typeinfo&)
lib.unexpected	17.3.2.14	unexpected()
lib.unreserved.names	17.1.5.9	Unreserved names
lib.uppercase	17.4.1.19	uppercase(ios&)
lib.wchar.t	17.2.2.1	Type wchar_t

lib.wint.t	17.3.1.4	Type `wint_t`
lib.ws	17.4.3.2	`ws(istream&)`
lib.wstring	17.5.2.1	Class `wstring`
lib.wstring::append.wc	17.5.2.1.13	`wstring::append(wchar_t, size_t)`
lib.wstring::append.wstr	17.5.2.1.12	`wstring::append(const wchar_t*, size_t)`
lib.wstring::append.wsub	17.5.2.1.11	`wstring::append(const wstring&, size_t, size_t)`
lib.wstring::assign.wc	17.5.2.1.16	`wstring::assign(wchar_t, size_t)`
lib.wstring::assign.wstr	17.5.2.1.15	`wstring::assign(const wchar_t*, size_t)`
lib.wstring::assign.wsub	17.5.2.1.14	`wstring::assign(const wstring&, size_t, size_t)`
lib.wstring::c.wcs	17.5.2.1.27	`wstring::c_wcs()`
lib.wstring::compare.wc	17.5.2.1.54	`wstring::compare(wchar_t, size_t)`
lib.wstring::compare.wstr	17.5.2.1.53	`wstring::compare(const wchar_t*, size_t)`
lib.wstring::compare.wsub	17.5.2.1.52	`wstring::compare(const wstring&, size_t, size_t)`
lib.wstring::copy.wstr	17.5.2.1.32	`wstring::copy(wchar_t*, size_t, size_t)`
lib.wstring::find.first.not.of.wc	17.5.2.1.47	`wstring::find_first_not_of(wchar_t, size_t)`
lib.wstring::find.first.not.of.wstr	17.5.2.1.46	`wstring::find_first_not_of(const wchar_t*, size_t, size_t)`
lib.wstring::find.first.not.of.wsub	17.5.2.1.45	`wstring::find_first_not_of(const wstring&, size_t)`
lib.wstring::find.first.of.wc	17.5.2.1.41	`wstring::find_first_of(wchar_t, size_t)`
lib.wstring::find.first.of.wstr	17.5.2.1.40	`wstring::find_first_of(const wchar_t*, size_t, size_t)`
lib.wstring::find.first.of.wsub	17.5.2.1.39	`wstring::find_first_of(const wstring&, size_t)`
lib.wstring::find.last.not.of.wc	17.5.2.1.50	`wstring::find_last_not_of(wchar_t, size_t)`
lib.wstring::find.last.not.of.wstr	17.5.2.1.49	`wstring::find_last_not_of(const wchar_t*, size_t, size_t)`
lib.wstring::find.last.not.of.wsub	17.5.2.1.48	`wstring::find_last_not_of(const wstring&, size_t)`
lib.wstring::find.last.of.wc	17.5.2.1.44	`wstring::find_last_of(wchar_t, size_t)`
lib.wstring::find.last.of.wstr	17.5.2.1.43	`wstring::find_last_of(const wchar_t*, size_t, size_t)`
lib.wstring::find.last.of.wsub	17.5.2.1.42	`wstring::find_last_of(const wstring&, size_t)`
lib.wstring::find.wc	17.5.2.1.35	`wstring::find(wchar_t, size_t)`
lib.wstring::find.wstr	17.5.2.1.34	`wstring::find(const wchar_t*, size_t, size_t)`
lib.wstring::find.wsub	17.5.2.1.33	`wstring::find(const wstring&, size_t)`
lib.wstring::get.at	17.5.2.1.24	`wstring::get_at(size_t)`

lib.wstring::insert.wc	17.5.2.1.19	wstring::insert(size_t, wchar_t, size_t)
lib.wstring::insert.wstr	17.5.2.1.18	wstring::insert(size_t, const wchar_t*, size_t)
lib.wstring::insert.wsub	17.5.2.1.17	wstring::insert(size_t, const wstring&, size_t, size_t)
lib.wstring::length	17.5.2.1.28	wstring::length()
lib.wstring::op+=.wc	17.5.2.1.10	wstring::operator+=(wchar_t)
lib.wstring::op+=.wstr	17.5.2.1.9	wstring::operator+=(const wchar_t*)
lib.wstring::op+=.wsub	17.5.2.1.8	wstring::operator+=(const wstring&)
lib.wstring::op.array	17.5.2.1.26	wstring::operator[](size_t)
lib.wstring::op=.wc	17.5.2.1.7	wstring::operator=(wchar_t)
lib.wstring::op=.wstr	17.5.2.1.6	wstring::operator=(const wchar_t*)
lib.wstring::put.at	17.5.2.1.25	wstring::put_at(size_t, wchar_t)
lib.wstring::remove	17.5.2.1.20	wstring::remove(size_t, size_t)
lib.wstring::replace.wc	17.5.2.1.23	wstring::replace(size_t, size_t, wchar_t, size_t)
lib.wstring::replace.wstr	17.5.2.1.22	wstring::replace(size_t, size_t, const wchar_t*, size_t)
lib.wstring::replace.wsub	17.5.2.1.21	wstring::replace(size_t, size_t, const wstring&, size_t, size_t)
lib.wstring::reserve	17.5.2.1.30	wstring::reserve()
lib.wstring::reserve.cap	17.5.2.1.31	wstring::reserve(size_t)
lib.wstring::resize	17.5.2.1.29	wstring::resize(size_t, wchar_t)
lib.wstring::rfind.wc	17.5.2.1.38	wstring::rfind(wchar_t, size_t)
lib.wstring::rfind.wstr	17.5.2.1.37	wstring::rfind(const wchar_t*, size_t, size_t)
lib.wstring::rfind.wsub	17.5.2.1.36	wstring::rfind(const wstring&, size_t)
lib.wstring::substr	17.5.2.1.51	wstring::substr(size_t, size_t)
lib.xalloc	17.3.3.1	Class xalloc
lib.xalloc::do.raise	17.3.3.1.3	xalloc::do_raise()
lib.xdomain	17.3.2.9	Class xdomain
lib.xdomain::do.raise	17.3.2.9.3	xdomain::do_raise()
lib.xlogic	17.3.2.2	Class xlogic
lib.xlogic::do.raise	17.3.2.2.3	xlogic::do_raise()
lib.xmsg	17.3.2.1	Class xmsg
lib.xmsg::do.raise	17.3.2.1.9	xmsg::do_raise()
lib.xmsg::raise	17.3.2.1.5	xmsg::raise()
lib.xmsg::raise.handler	17.3.2.1.1	Type xmsg::raise_handler
lib.xmsg::set.raise.handler	17.3.2.1.2	xmsg::set_raise_handler(raise_handler)
lib.xmsg::what	17.3.2.1.6	xmsg::what()
lib.xmsg::where	17.3.2.1.7	xmsg::where()
lib.xmsg::why	17.3.2.1.8	xmsg::why()
lib.xrange	17.3.2.10	Class xrange
lib.xrange::do.raise	17.3.2.10.3	xrange::do_raise()
lib.xruntime	17.3.2.3	Class xruntime
lib.xruntime::do.raise	17.3.2.3.3	xruntime::do_raise()
limits	B	Implementation quantities
namespace.alias	7.3.2	Namespace or class alias
namespace.def	7.3.1	Namespace definition
namespace.qual	7.3.5	Explict qualification
namespace.udecl	7.3.3	The using declaration
namespace.udir	7.3.4	Using directive
over	13	Overloading

over.ass	13.4.3	Assignment
over.binary	13.4.2	Binary operators
over.call	13.4.4	Function call
over.dcl	13.1	Declaration matching
over.inc	13.4.7	Increment and decrement
over.match	13.2	Overload resolution
over.match.args	13.2.2	Argument matching
over.match.funcs	13.2.1	Candidate functions
over.oper	13.4	Overloaded operators
over.oper.funcs	13.4.8	Overloaded operators in expressions
over.over	13.3	Address of overloaded function
over.ref	13.4.6	Class member access
over.sub	13.4.5	Subscripting
over.unary	13.4.1	Unary operators
special	12	Special member functions
stmt.ambig	6.8	Ambiguity resolution
stmt.block	6.3	Compound statement or block
stmt.break	6.6.1	The `break` statement
stmt.cont	6.6.2	The `continue` statement
stmt.dcl	6.7	Declaration statement
stmt.do	6.5.2	The `do` statement
stmt.expr	6.2	Expression statement
stmt.for	6.5.3	The `for` statement
stmt.goto	6.6.4	The `goto` statement
stmt.if	6.4.1	The `if` statement
stmt.iter	6.5	Iteration statements
stmt.jump	6.6	Jump statements
stmt.label	6.1	Labeled statement
stmt.return	6.6.3	The `return` statement
stmt.select	6.4	Selection statements
stmt.stmt	6	Statements
stmt.switch	6.4.2	The `switch` statement
stmt.while	6.5.1	The `while` statement
syntax	1.4	Syntax notation
temp	14	Templates
temp.arg	14.7	Template arguments
temp.arg.explicit	14.9.1	Explicit template argument specification
temp.deduct	14.9.2	Template argument deduction
temp.dep	14.2.3	Dependent names
temp.encl	14.2.2	Names from the template's enclosing scope
temp.explicit	14.4	Explicit instantiation
temp.fct	14.9	Function templates
temp.friend	14.11	Friends
temp.inject	14.2.4	Non-local names declared within a template
temp.inst	14.3	Template instantiation
temp.local	14.2.1	Locally declared names
temp.mem.func	14.10	Member function templates
temp.names	14.1	Template names
temp.over	14.9.3	Overload resolution
temp.over.spec	14.9.4	Overloading and specialization
temp.param	14.6	Template parameters
temp.res	14.2	Name resolution
temp.spec	14.5	Template specialization
temp.static	14.12	Static members and variables
temp.type	14.8	Type equivalence