

Index

- [*x*], **3.18**
- [*x*], **3.19**
- ! (logical negation operator), **6.5.3.3**
- != (inequality operator), **6.5.9**
- # operator, **6.10.3.2**
- # preprocessing directive, **6.10.7**
- # punctuator, **6.10**
- ## operator, **6.10.3.3**
- #define preprocessing directive, **6.10.3**
- #elif preprocessing directive, **6.10.1**
- #else preprocessing directive, **6.10.1**
- #endif preprocessing directive, **6.10.1**
- #error preprocessing directive, **4, 6.10.5**
- #if preprocessing directive, **5.2.4.2.1, 5.2.4.2.2, 6.10.1, 7.1.4**
- #ifdef preprocessing directive, **6.10.1**
- #ifndef preprocessing directive, **6.10.1**
- #include preprocessing directive, **5.1.1.2, 6.10.2**
- #line preprocessing directive, **6.10.4**
- #pragma preprocessing directive, **6.10.6**
- #undef preprocessing directive, **6.10.3.5, 7.1.3, 7.1.4**
- % (remainder operator), **6.5.5**
- #: (alternative spelling of #), **6.4.6**
- %%: (alternative spelling of ##), **6.4.6**
- %= (remainder assignment operator), **6.5.16.2**
- %> (alternative spelling of }), **6.4.6**
- & (address operator), **6.3.2.1, 6.5.3.2**
- & (bitwise AND operator), **6.5.10**
- && (logical AND operator), **6.5.13**
- &= (bitwise AND assignment operator), **6.5.16.2**
- ' ' (space character), **5.1.1.2, 5.2.1, 6.4, 7.4.1.3, 7.4.1.10, 7.25.2.1.3**
- () (cast operator), **6.5.4**
- () (function-call operator), **6.5.2.2**
- () (parentheses punctuator), **6.7.5.3, 6.8.4, 6.8.5**
- () { } (compound-literal operator), **6.5.2.5**
- * (asterisk punctuator), **6.7.5.1, 6.7.5.2**
- * (indirection operator), **6.5.2.1, 6.5.3.2**
- * (multiplication operator), **6.5.5, F.3, G.5.1**
- *= (multiplication assignment operator), **6.5.16.2**
- + (addition operator), **6.5.2.1, 6.5.3.2, 6.5.6, F.3, G.5.2**
- + (unary plus operator), **6.5.3.3**
- ++ (postfix increment operator), **6.3.2.1, 6.5.2.4**
- ++ (prefix increment operator), **6.3.2.1, 6.5.3.1**
- += (addition assignment operator), **6.5.16.2**
- , (comma operator), **6.5.17**
- , (comma punctuator), **6.5.2, 6.7, 6.7.2.1, 6.7.2.2, 6.7.2.3, 6.7.8**
- (subtraction operator), **6.5.6, F.3, G.5.2**
- (unary minus operator), **6.5.3.3, F.3**
- (postfix decrement operator), **6.3.2.1, 6.5.2.4**
- (prefix decrement operator), **6.3.2.1, 6.5.3.1**
- = (subtraction assignment operator), **6.5.16.2**
- > (structure/union pointer operator), **6.5.2.3**
- . (structure/union member operator), **6.3.2.1, 6.5.2.3**
- . punctuator, **6.7.8**
- ... (ellipsis punctuator), **6.5.2.2, 6.7.5.3, 6.10.3**
- / (division operator), **6.5.5, F.3, G.5.1**
- /* */ (comment delimiters), **6.4.9**
- // (comment delimiter), **6.4.9**
- /= (division assignment operator), **6.5.16.2**
- : (colon punctuator), **6.7.2.1**
- :> (alternative spelling of]), **6.4.6**
- ; (semicolon punctuator), **6.7, 6.7.2.1, 6.8.3, 6.8.5, 6.8.6**
- < (less-than operator), **6.5.8**
- <% (alternative spelling of {), **6.4.6**
- <: (alternative spelling of [), **6.4.6**
- << (left-shift operator), **6.5.7**
- <<= (left-shift assignment operator), **6.5.16.2**
- <= (less-than-or-equal-to operator), **6.5.8**
- <assert.h> header, **7.2, B.1**
- <complex.h> header, **5.2.4.2.2, 7.3, 7.22, 7.26.1, G.6, J.5.17**
- <ctype.h> header, **7.4, 7.26.2**
- <errno.h> header, **7.5, 7.26.3**
- <fenv.h> header, **5.1.2.3, 5.2.4.2.2, 7.6, 7.12, F, H**
- <float.h> header, **4, 5.2.4.2.2, 7.7, 7.20.1.3, 7.24.4.1.1**
- <inttypes.h> header, **7.8, 7.26.4**
- <iso646.h> header, **4, 7.9**
- <limits.h> header, **4, 5.2.4.2.1, 6.2.5, 7.10**
- <locale.h> header, **7.11, 7.26.5**
- <math.h> header, **5.2.4.2.2, 6.5, 7.12, 7.22, F, F.9, J.5.17**
- <setjmp.h> header, **7.13**
- <signal.h> header, **7.14, 7.26.6**
- <stdarg.h> header, **4, 6.7.5.3, 7.15**
- <stdbool.h> header, **4, 7.16, 7.26.7, H**
- <stddef.h> header, **4, 6.3.2.1, 6.3.2.3, 6.4.4.4, 6.4.5, 6.5.3.4, 6.5.6, 7.17**
- <stdint.h> header, **4, 5.2.4.2, 6.10.1, 7.8, 7.18, 7.26.8**

- <stdio.h>** header, 7.19, 7.26.9, F
<stdlib.h> header, 7.20, 7.26.10, F
<string.h> header, 7.21, 7.26.11
<tgmath.h> header, 7.22, G.7
<time.h> header, 7.23
<wchar.h> header, 7.19.1, 7.24, 7.26.12, F
<wctype.h> header, 7.25, 7.26.13
 = (equal-sign punctuation), 6.7, 6.7.2.2, 6.7.8
 = (simple assignment operator), 6.5.16.1
 == (equality operator), 6.5.9
 > (greater-than operator), 6.5.8
 >= (greater-than-or-equal-to operator), 6.5.8
 >> (right-shift operator), 6.5.7
 >>= (right-shift assignment operator), 6.5.16.2
 ? : (conditional operator), 6.5.15
 ?? (trigraph sequences), 5.2.1.1
 [] (array subscript operator), 6.5.2.1, 6.5.3.2
 [] (brackets punctuation), 6.7.5.2, 6.7.8
 \ (backslash character), 5.1.1.2, 5.2.1, 6.4.4.4
 \ (escape character), 6.4.4.4
 \" (double-quote escape sequence), 6.4.4.4, 6.4.5, 6.10.9
 \\ (backslash escape sequence), 6.4.4.4, 6.10.9
 \' (single-quote escape sequence), 6.4.4.4, 6.4.5
 \0 (null character), 5.2.1, 6.4.4.4, 6.4.5
 padding of binary stream, 7.19.2
 \? (question-mark escape sequence), 6.4.4.4
 \a (alert escape sequence), 5.2.2, 6.4.4.4
 \b (backspace escape sequence), 5.2.2, 6.4.4.4
 \f (form-feed escape sequence), 5.2.2, 6.4.4.4, 7.4.1.10
 \n (new-line escape sequence), 5.2.2, 6.4.4.4, 7.4.1.10
 \octal digits (octal-character escape sequence), 6.4.4.4
 \r (carriage-return escape sequence), 5.2.2, 6.4.4.4, 7.4.1.10
 \t (horizontal-tab escape sequence), 5.2.2, 6.4.4.4, 7.4.1.3, 7.4.1.10, 7.25.2.1.3
 \U (universal character names), 6.4.3
 \u (universal character names), 6.4.3
 \v (vertical-tab escape sequence), 5.2.2, 6.4.4.4, 7.4.1.10
 \x hexadecimal digits (hexadecimal-character escape sequence), 6.4.4.4
 ^ (bitwise exclusive OR operator), 6.5.11
 ^= (bitwise exclusive OR assignment operator), 6.5.16.2
 __bool_true_false_are_defined macro, 7.16
 __cplusplus macro, 6.10.8
 __DATE__ macro, 6.10.8
 __FILE__ macro, 6.10.8, 7.2.1.1
 __func__ identifier, 6.4.2.2, 7.2.1.1
 __LINE__ macro, 6.10.8, 7.2.1.1
 __STDC__, 6.11.9
 __STDC__ macro, 6.10.8
 __STDC_CONSTANT_MACROS macro, 7.18.4
 __STDC_FORMAT_MACROS macro, 7.8.1
 __STDC_HOSTED__ macro, 6.10.8
 __STDC_IEC_559__ macro, 6.10.8, F.1
 __STDC_IEC_559_COMPLEX__ macro, 6.10.8, G.1
 __STDC_ISO_10646__ macro, 6.10.8
 __STDC_LIMIT_MACROS macro, 7.18.2, 7.18.3
 __STDC_VERSION__ macro, 6.10.8
 __TIME__ macro, 6.10.8
 __VA_ARGS__ identifier, 6.10.3, 6.10.3.1
 _Bool type, 6.2.5, 6.3.1.1, 6.3.1.2, 6.7.2
 _Bool type conversions, 6.3.1.2
 _Complex types, 6.2.5, 6.7.2, 7.3.1, G
 _Complex_I macro, 7.3.1
 _Exit function, 7.20.4.4
 _Imaginary types, 6.7.2, 7.3.1, G
 _Imaginary_I macro, 7.3.1, G.6
 _IOFBF macro, 7.19.1, 7.19.5.5, 7.19.5.6
 _IOLBF macro, 7.19.1, 7.19.5.6
 _IONBF macro, 7.19.1, 7.19.5.5, 7.19.5.6
 _Pragma operator, 5.1.1.2, 6.10.9
 { } (braces punctuation), 6.7.2.2, 6.7.2.3, 6.7.8, 6.8.2
 { } (compound-literal operator), 6.5.2.5
 | (bitwise inclusive OR operator), 6.5.12
 |= (bitwise inclusive OR assignment operator), 6.5.16.2
 || (logical OR operator), 6.5.14
 ~ (bitwise complement operator), 6.5.3.3
 abort function, 7.2.1.1, 7.14.1.1, 7.19.3, 7.20.4.1
 abs function, 7.20.6.1
 absolute-value functions
 complex, 7.3.8, G.6.4
 integer, 7.8.2.1, 7.20.6.1
 real, 7.12.7, F.9.4
 abstract declarator, 6.7.6
 abstract machine, 5.1.2.3
 access, 3.1, 6.7.3
 accuracy, *see* floating-point accuracy
 acos functions, 7.12.4.1, F.9.1.1
 acos type-generic macro, 7.22

- acosh** functions, **7.12.5.1, F.9.2.1**
- acosh** type-generic macro, **7.22**
- active position, **5.2.2**
- actual argument, **3.3**
- actual parameter (deprecated), **3.3**
- addition assignment operator (**+=**), **6.5.16.2**
- addition operator (**+**), **6.5.2.1, 6.5.3.2, 6.5.6, F.3, G.5.2**
- additive expressions, **6.5.6, G.5.2**
- address constant, **6.6**
- address operator (**&**), **6.3.2.1, 6.5.3.2**
- aggregate initialization, **6.7.8**
- aggregate types, **6.2.5**
- alert escape sequence (**\a**), **5.2.2, 6.4.4.4**
- aliasing, **6.5**
- alignment, **3.2**
 - pointer, **6.2.5, 6.3.2.3**
 - structure/union member, **6.7.2.1**
- allocated storage, order and contiguity, **7.20.3**
- and** macro, **7.9**
- AND operators
 - bitwise (**&**), **6.5.10**
 - bitwise assignment (**&=**), **6.5.16.2**
 - logical (**&&**), **6.5.13**
- and_eq** macro, **7.9**
- ANSI/IEEE 754, **F.1**
- ANSI/IEEE 854, **F.1**
- argc** (**main** function parameter), **5.1.2.2.1**
- argument, **3.3**
 - array, **6.9.1**
 - default promotions, **6.5.2.2**
 - function, **6.5.2.2, 6.9.1**
 - macro, substitution, **6.10.3.1**
- argument, complex, **7.3.9.1**
- argv** (**main** function parameter), **5.1.2.2.1**
- arithmetic constant expression, **6.6**
- arithmetic conversions, usual, *see* usual arithmetic conversions
- arithmetic operators
 - additive, **6.5.6, G.5.2**
 - bitwise, **6.5.10, 6.5.11, 6.5.12**
 - increment and decrement, **6.5.2.4, 6.5.3.1**
 - multiplicative, **6.5.5, G.5.1**
 - shift, **6.5.7**
 - unary, **6.5.3.3**
- arithmetic types, **6.2.5**
- arithmetic, pointer, **6.5.6**
- array
 - argument, **6.9.1**
 - declarator, **6.7.5.2**
 - initialization, **6.7.8**
 - multidimensional, **6.5.2.1**
 - parameter, **6.9.1**
 - storage order, **6.5.2.1**
 - subscript operator (**[]**), **6.5.2.1, 6.5.3.2**
 - subscripting, **6.5.2.1**
 - type, **6.2.5**
 - type conversion, **6.3.2.1**
- arrow operator (**->**), **6.5.2.3**
- as-if rule, **5.1.2.3**
- ASCII code set, **5.2.1.1**
- asctime** function, **7.23.3.1**
- asin** functions, **7.12.4.2, F.9.1.2**
- asin** type-generic macro, **7.22, G.7**
- asinh** functions, **7.12.5.2, F.9.2.2**
- asinh** type-generic macro, **7.22, G.7**
- asm** keyword, **J.5.10**
- assert** macro, **7.2.1.1**
- assert.h** header, **7.2, B.1**
- assignment
 - compound, **6.5.16.2**
 - conversion, **6.5.16.1**
 - expression, **6.5.16**
 - operators, **6.3.2.1, 6.5.16**
 - simple, **6.5.16.1**
- associativity of operators, **6.5**
- asterisk punctuator (*****), **6.7.5.1, 6.7.5.2**
- atan** functions, **7.12.4.3, F.9.1.3**
- atan** type-generic macro, **7.22, G.7**
- atan2** functions, **7.12.4.4, F.9.1.4**
- atan2** type-generic macro, **7.22**
- atanh** functions, **7.12.5.3, F.9.2.3**
- atanh** type-generic macro, **7.22, G.7**
- atexit** function, **7.20.4.2, 7.20.4.3, 7.20.4.4, J.5.13**
- atof** function, **7.20.1, 7.20.1.1**
- atoi** function, **7.20.1, 7.20.1.2**
- atol** function, **7.20.1, 7.20.1.2**
- atoll** function, **7.20.1, 7.20.1.2**
- auto** storage-class specifier, **6.7.1, 6.9**
- automatic storage duration, **5.2.3, 6.2.4**
- backslash character (****), **5.1.1.2, 5.2.1, 6.4.4.4**
- backslash escape sequence (****), **6.4.4.4, 6.10.9**
- backspace escape sequence (**\b**), **5.2.2, 6.4.4.4**
- basic character set, **3.6, 3.7.2, 5.2.1**
- basic types, **6.2.5**
- behavior, **3.4**
- binary streams, **7.19.2, 7.19.7.11, 7.19.9.2, 7.19.9.4**
- bit, **3.5**
 - high order, **3.6**

- low order, 3.6
- bit-field, 6.7.2.1
- bitand** macro, 7.9
- bitor** macro, 7.9
- bitwise operators, 6.5
 - AND, 6.5.10
 - AND assignment (&=), 6.5.16.2
 - complement (~), 6.5.3.3
 - exclusive OR, 6.5.11
 - exclusive OR assignment (^=), 6.5.16.2
 - inclusive OR, 6.5.12
 - inclusive OR assignment (|=), 6.5.16.2
 - shift, 6.5.7
- blank character, 7.4.1.3
- block, 6.8, 6.8.2, 6.8.4, 6.8.5
- block scope, 6.2.1
- block structure, 6.2.1
- bold type** convention, 6.1
- bool** macro, 7.16
- boolean type, 6.3.1.2
- boolean type conversion, 6.3.1.1, 6.3.1.2
- braces punctuator ({ }), 6.7.2.2, 6.7.2.3, 6.7.8, 6.8.2
- brackets operator ([]), 6.5.2.1, 6.5.3.2
- brackets punctuator ([]), 6.7.5.2, 6.7.8
- branch cuts, 7.3.3
- break** statement, 6.8.6.3
- broken-down time, 7.23.1, 7.23.2.3, 7.23.3, 7.23.3.1, 7.23.3.3, 7.23.3.4, 7.23.3.5
- bsearch** function, 7.20.5, 7.20.5.1
- btowc** function, 7.24.6.1.1
- BUFSIZ** macro, 7.19.1, 7.19.2, 7.19.5.5
- byte, 3.6, 6.5.3.4
- byte input/output functions, 7.19.1
- byte-oriented stream, 7.19.2
- C program, 5.1.1.1
- C++, 7.8.1, 7.18.2, 7.18.3, 7.18.4
- cabs** functions, 7.3.8.1, G.6
 - type-generic macro for, 7.22
- acos** functions, 7.3.5.1, G.6.1.1
 - type-generic macro for, 7.22
- acosh** functions, 7.3.6.1, G.6.2.1
 - type-generic macro for, 7.22
- calendar time, 7.23.1, 7.23.2.2, 7.23.2.3, 7.23.2.4, 7.23.3.2, 7.23.3.3, 7.23.3.4
- call by value, 6.5.2.2
- calloc** function, 7.20.3, 7.20.3.1, 7.20.3.2, 7.20.3.4
- carg** functions, 7.3.9.1, G.6
- carg** type-generic macro, 7.22, G.7
- carriage-return escape sequence (\r), 5.2.2, 6.4.4.4, 7.4.1.10
- case** label, 6.8.1, 6.8.4.2
- case mapping functions
 - character, 7.4.2
 - wide character, 7.25.3.1
 - extensible, 7.25.3.2
- casin** functions, 7.3.5.2, G.6
 - type-generic macro for, 7.22
- casinh** functions, 7.3.6.2, G.6.2.2
 - type-generic macro for, 7.22
- cast expression, 6.5.4
- cast operator (()), 6.5.4
- catan** functions, 7.3.5.3, G.6
 - type-generic macro for, 7.22
- catanh** functions, 7.3.6.3, G.6.2.3
 - type-generic macro for, 7.22
- cbt** functions, 7.12.7.1, F.9.4.1
- cbt** type-generic macro, 7.22
- ccos** functions, 7.3.5.4, G.6
 - type-generic macro for, 7.22
- ccosh** functions, 7.3.6.4, G.6.2.4
 - type-generic macro for, 7.22
- ceil** functions, 7.12.9.1, F.9.6.1
- ceil** type-generic macro, 7.22
- cerf** function, 7.26.1
- cerfc** function, 7.26.1
- cexp** functions, 7.3.7.1, G.6.3.1
 - type-generic macro for, 7.22
- cexp2** function, 7.26.1
- cexpm1** function, 7.26.1
- char** type, 6.2.5, 6.3.1.1, 6.7.2
- char** type conversion, 6.3.1.1, 6.3.1.3, 6.3.1.4, 6.3.1.8
- CHAR_BIT** macro, 5.2.4.2.1
- CHAR_MAX** macro, 5.2.4.2.1, 7.11.2.1
- CHAR_MIN** macro, 5.2.4.2.1
- character, 3.7, 3.7.1
- character array initialization, 6.7.8
- character case mapping functions, 7.4.2
 - wide character, 7.25.3.1
 - extensible, 7.25.3.2
- character classification functions, 7.4.1
 - wide character, 7.25.2.1
 - extensible, 7.25.2.2
- character constant, 5.1.1.2, 5.2.1, 6.4.4.4
- character display semantics, 5.2.2
- character handling header, 7.4, 7.11.1.1
- character input/output functions, 7.19.7
 - wide character, 7.24.3
- character sets, 5.2.1

- character string literal, *see* string literal
- character type conversion, **6.3.1.1**
- character types, **6.2.5**, 6.7.8
- cimag** functions, **7.3.9.2**, 7.3.9.4, **G.6**
- cimag** type-generic macro, **7.22**, **G.7**
- cis function, **G.6**
- classification functions
 - character, **7.4.1**
 - wide character, **7.25.2.1**
 - extensible, **7.25.2.2**
- clearerr** function, **7.19.10.1**
- clgamma** function, **7.26.1**
- clock** function, **7.23.2.1**
- clock_t** type, **7.23.1**, 7.23.2.1
- CLOCKS_PER_SEC** macro, **7.23.1**, 7.23.2.1
- clog** functions, **7.3.7.2**, **G.6.3.2**
 - type-generic macro for, **7.22**
- clog10** function, **7.26.1**
- clog1p** function, **7.26.1**
- clog2** function, **7.26.1**
- collating sequences, **5.2.1**
- colon punctuator (**:**), 6.7.2.1
- comma operator (**,**), **6.5.17**
- comma punctuator (**,**), 6.5.2, 6.7, 6.7.2.1, 6.7.2.2, 6.7.2.3, 6.7.8
- command processor, **7.20.4.6**
- comment delimiters (**/* */** and **//**), **6.4.9**
- comments, 5.1.1.2, 6.4, **6.4.9**
- common extensions, **J.5**
- common initial sequence, **6.5.2.3**
- common real type, **6.3.1.8**
- common warnings, **I**
- comparison functions, **7.20.5**, 7.20.5.1, 7.20.5.2
 - string, **7.21.4**
 - wide string, **7.24.4.4**
- comparison macros, **7.12.14**
- comparison, pointer, **6.5.8**
- compatible type, **6.2.7**, 6.7.2, 6.7.3, 6.7.5
- compl** macro, **7.9**
- complement operator (**~**), **6.5.3.3**
- complex** macro, **7.3.1**
- complex numbers, **6.2.5**, **G**
- complex type conversion, **6.3.1.6**, 6.3.1.7
- complex type domain, **6.2.5**
- complex types, **6.2.5**, 6.7.2, **G**
- complex.h** header, 5.2.4.2.2, **7.3**, 7.22, 7.26.1, **G.6**, **J.5.17**
- compliance, *see* conformance
- components of time, **7.23.1**
- composite type, **6.2.7**
- compound assignment, **6.5.16.2**
- compound literals, **6.5.2.5**
- compound statement, **6.8.2**
- compound-literal operator (**(() { })**), **6.5.2.5**
- concatenation functions
 - string, **7.21.3**
 - wide string, **7.24.4.3**
- concatenation, preprocessing, *see* preprocessing concatenation
- conceptual models, **5.1**
- conditional inclusion, **6.10.1**
- conditional operator (**? :**), **6.5.15**
- conformance, **4**
- conj** functions, **7.3.9.3**, **G.6**
- conj** type-generic macro, **7.22**
- const** type qualifier, **6.7.3**
- const-qualified type, **6.2.5**, 6.3.2.1, 6.7.3
- constant expression, **6.6**, F.7.4
- constants, **6.4.4**
 - as primary expression, **6.5.1**
 - character, **6.4.4.4**
 - enumeration, **6.2.1**, **6.4.4.3**
 - floating, **6.4.4.2**
 - hexadecimal, **6.4.4.1**
 - integer, **6.4.4.1**
 - octal, **6.4.4.1**
- constraint, **3.8**, **4**
- content of structure/union/enumeration, **6.7.2.3**
- contiguity of allocated storage, **7.20.3**
- continue** statement, **6.8.6.2**
- contracted expression, **6.5**, 7.12.2, **F.6**
- control character, 5.2.1, **7.4**
- control wide character, **7.25.2**
- conversion, **6.3**
 - arithmetic operands, **6.3.1**
 - array, **6.3.2.1**
 - array argument, **6.9.1**
 - array parameter, **6.9.1**
 - arrays, **6.3.2.1**
 - boolean, **6.3.1.2**
 - boolean, characters, and integers, **6.3.1.1**
 - by assignment, **6.5.16.1**
 - by **return** statement, **6.8.6.4**
 - complex types, **6.3.1.6**
 - explicit, **6.3**
 - function, **6.3.2.1**
 - function argument, 6.5.2.2, **6.9.1**
 - function designators, **6.3.2.1**
 - function parameter, **6.9.1**
 - imaginary, **G.4.1**
 - imaginary and complex, **G.4.3**
 - implicit, **6.3**

- lvalues, **6.3.2.1**
- pointer, 6.3.2.1, **6.3.2.3**
- real and complex, **6.3.1.7**
- real and imaginary, **G.4.2**
- real floating and integer, **6.3.1.4**, F.3, F.4
- real floating types, **6.3.1.5**, F.3
- signed and unsigned integers, **6.3.1.3**
- usual arithmetic, *see* usual arithmetic conversions
- void** type, **6.3.2.2**
- conversion functions
 - multibyte/wide character, **7.20.7**
 - extended, **7.24.6**
 - restartable, **7.24.6.3**
 - multibyte/wide string, **7.20.8**
 - restartable, **7.24.6.4**
 - numeric, **7.8.2.3**, **7.20.1**
 - wide string, **7.8.2.4**, **7.24.4.1**
 - single byte/wide character, **7.24.6.1**
 - time, **7.23.3**
 - wide character, **7.24.5**
- conversion specifier, **7.19.6.1**, **7.19.6.2**, **7.24.2.1**, **7.24.2.2**
- conversion state, **7.20.7**, **7.24.6**, 7.24.6.2.1, 7.24.6.3, 7.24.6.3.2, 7.24.6.3.3, 7.24.6.4, 7.24.6.4.1, 7.24.6.4.2
- conversion state functions, **7.24.6.2**
- copying functions
 - string, **7.21.2**
 - wide string, **7.24.4.2**
- copysign** functions, 7.3.9.4, **7.12.11.1**, F.3, **F.9.8.1**
- copysign** type-generic macro, **7.22**
- correctly rounded result, **3.9**
- corresponding real type, **6.2.5**
- cos** functions, **7.12.4.5**, **F.9.1.5**
- cos** type-generic macro, **7.22**, **G.7**
- cosh** functions, **7.12.5.4**, **F.9.2.4**
- cosh** type-generic macro, **7.22**, **G.7**
- cpow** functions, **7.3.8.2**, **G.6.4.1**
 - type-generic macro for, **7.22**
- cproj** functions, **7.3.9.4**, **G.6**
- cproj** type-generic macro, **7.22**
- creal** functions, **7.3.9.5**, **G.6**
- creal** type-generic macro, **7.22**, **G.7**
- csin** functions, **7.3.5.5**, **G.6**
 - type-generic macro for, **7.22**
- csinh** functions, **7.3.6.5**, **G.6.2.5**
 - type-generic macro for, **7.22**
- csqrt** functions, **7.3.8.3**, **G.6.4.2**
 - type-generic macro for, **7.22**
- ctan** functions, **7.3.5.6**, **G.6**
 - type-generic macro for, **7.22**
- ctanh** functions, **7.3.6.6**, **G.6.2.6**
 - type-generic macro for, **7.22**
- ctgamma** function, **7.26.1**
- ctime** function, **7.23.3.2**
- ctype.h** header, **7.4**, 7.26.2
- current object, **6.7.8**
- CX_LIMITED_RANGE** pragma, 6.10.6, **7.3.4**
- data stream, *see* streams
- date and time header, **7.23**
- Daylight Saving Time, **7.23.1**
- DBL_DIG** macro, **5.2.4.2.2**
- DBL_EPSILON** macro, **5.2.4.2.2**
- DBL_MANT_DIG** macro, **5.2.4.2.2**
- DBL_MAX** macro, **5.2.4.2.2**
- DBL_MAX_10_EXP** macro, **5.2.4.2.2**
- DBL_MAX_EXP** macro, **5.2.4.2.2**
- DBL_MIN** macro, **5.2.4.2.2**
- DBL_MIN_10_EXP** macro, **5.2.4.2.2**
- DBL_MIN_EXP** macro, **5.2.4.2.2**
- decimal constant, **6.4.4.1**
- decimal digit, **5.2.1**
- decimal-point character, **7.1.1**, 7.11.2.1
- DECIMAL_DIG** macro, **5.2.4.2.2**, 7.19.6.1, 7.20.1.3, 7.24.2.1, 7.24.4.1.1, F.5
- declaration specifiers, **6.7**
- declarations, **6.7**
 - function, **6.7.5.3**
 - pointer, **6.7.5.1**
 - structure/union, **6.7.2.1**
 - typedef**, **6.7.7**
- declarator, **6.7.5**
 - abstract, **6.7.6**
- declarator type derivation, **6.2.5**, 6.7.5
- decrement operators, *see* arithmetic operators, increment and decrement
- default argument promotions, **6.5.2.2**
- default initialization, **6.7.8**
- default** label, 6.8.1, **6.8.4.2**
- define** preprocessing directive, **6.10.3**
- defined** operator, **6.10.1**, 6.10.8
- definition, **6.7**
 - function, **6.9.1**
- derived declarator types, **6.2.5**
- derived types, **6.2.5**
- designated initializer, **6.7.8**
- destringizing, **6.10.9**
- device input/output, **5.1.2.3**
- diagnostic message, **3.10**, 5.1.1.3

- diagnostics, **5.1.1.3**
- diagnostics header, **7.2**
- difftime** function, **7.23.2.2**
- digit, **5.2.1**, 7.4
- digraphs, **6.4.6**
- direct input/output functions, **7.19.8**
- display device, **5.2.2**
- div** function, **7.20.6.2**
- div_t** type, **7.20**
- division assignment operator (/=), **6.5.16.2**
- division operator (/), **6.5.5**, F.3, G.5.1
- do** statement, **6.8.5.2**
- documentation of implementation, **4**
- domain error, **7.12.1**, 7.12.4.1, 7.12.4.2, 7.12.4.4, 7.12.5.1, 7.12.5.3, 7.12.6.7, 7.12.6.8, 7.12.6.9, 7.12.6.10, 7.12.6.11, 7.12.7.4, 7.12.7.5, 7.12.8.4, 7.12.10.1
- dot operator (.), **6.5.2.3**
- double _Complex** type, **6.2.5**
- double _Complex** type conversion, **6.3.1.6**, **6.3.1.7**, **6.3.1.8**
- double _Imaginary** type, **G.2**
- double** type, **6.2.5**, 6.4.4.2, 6.7.2, 7.19.6.2, 7.24.2.2, F.2
- double** type conversion, **6.3.1.4**, **6.3.1.5**, **6.3.1.7**, **6.3.1.8**
- double-precision arithmetic, **5.1.2.3**
- double-quote escape sequence (\"), **6.4.4.4**, **6.4.5**, 6.10.9
- double_t** type, **7.12**, J.5.6
- EDOM** macro, **7.5**, 7.12.1, *see also* domain error
- effective type, **6.5**
- EILSEQ** macro, **7.5**, 7.19.3, 7.24.3.1, 7.24.3.3, 7.24.6.3.2, 7.24.6.3.3, 7.24.6.4.1, 7.24.6.4.2, *see also* encoding error
- element type, **6.2.5**
- elif** preprocessing directive, **6.10.1**
- ellipsis punctuator (...), 6.5.2.2, **6.7.5.3**, **6.10.3**
- else** preprocessing directive, **6.10.1**
- else** statement, **6.8.4.1**
- empty statement, **6.8.3**
- encoding error, **7.19.3**, 7.24.3.1, 7.24.3.3, 7.24.6.3.2, 7.24.6.3.3, 7.24.6.4.1, 7.24.6.4.2
- end-of-file, **7.24.1**
- end-of-file indicator, **7.19.1**, 7.19.5.3, 7.19.7.1, 7.19.7.5, 7.19.7.6, 7.19.7.11, 7.19.9.2, 7.19.9.3, 7.19.10.1, 7.19.10.2, 7.24.3.1, 7.24.3.10
- end-of-file macro, *see* **EOF** macro
- end-of-line indicator, **5.2.1**
- endif** preprocessing directive, **6.10.1**
- enum** type, **6.2.5**, 6.7.2, **6.7.2.2**
- enumerated type, **6.2.5**
- enumeration, **6.2.5**, 6.7.2.2
- enumeration constant, **6.2.1**, **6.4.4.3**
- enumeration content, **6.7.2.3**
- enumeration members, **6.7.2.2**
- enumeration specifiers, **6.7.2.2**
- enumeration tag, 6.2.3, **6.7.2.3**
- enumerator, **6.7.2.2**
- environment, **5**
- environment functions, **7.20.4**
- environment list, **7.20.4.5**
- environmental considerations, **5.2**
- environmental limits, **5.2.4**, 7.13.1.1, 7.19.2, 7.19.3, 7.19.4.4, 7.19.6.1, 7.20.2.1, 7.20.4.2, 7.24.2.1
- EOF** macro, 7.4, **7.19.1**, 7.19.5.1, 7.19.5.2, 7.19.6.2, 7.19.6.7, 7.19.6.9, 7.19.6.11, 7.19.6.14, 7.19.7.1, 7.19.7.3, 7.19.7.4, 7.19.7.5, 7.19.7.6, 7.19.7.9, 7.19.7.10, 7.19.7.11, 7.24.1, 7.24.2.2, 7.24.2.4, 7.24.2.6, 7.24.2.8, 7.24.2.10, 7.24.2.12, 7.24.3.4, 7.24.6.1.1, 7.24.6.1.2
- equal-sign punctuator (=), 6.7, 6.7.2.2, 6.7.8
- equal-to operator, *see* equality operator
- equality expressions, **6.5.9**
- equality operator (==), **6.5.9**
- ERANGE** macro, **7.5**, 7.8.2.3, 7.8.2.4, 7.12.1, 7.20.1.3, 7.20.1.4, 7.24.4.1.1, 7.24.4.1.2, *see also* range error
- erf** functions, **7.12.8.1**, **F.9.5.1**
- erf** type-generic macro, **7.22**
- erfc** functions, **7.12.8.2**, **F.9.5.2**
- erfc** type-generic macro, **7.22**
- errno** macro, 7.1.3, 7.3.2, **7.5**, 7.8.2.3, 7.8.2.4, 7.12.1, 7.14.1.1, 7.19.3, 7.19.9.3, 7.19.10.4, 7.20.1, 7.20.1.3, 7.20.1.4, 7.21.6.2, 7.24.3.1, 7.24.3.3, 7.24.4.1.1, 7.24.4.1.2, 7.24.6.3.2, 7.24.6.3.3, 7.24.6.4.1, 7.24.6.4.2, J.5.17
- errno.h** header, **7.5**, 7.26.3
- error
 - domain, *see* domain error
 - encoding, *see* encoding error
 - range, *see* range error
- error conditions, **7.12.1**
- error functions, **7.12.8**, **F.9.5**
- error indicator, **7.19.1**, 7.19.5.3, 7.19.7.1, 7.19.7.3, 7.19.7.5, 7.19.7.6, 7.19.7.8, 7.19.7.9, 7.19.9.2, 7.19.10.1, 7.19.10.3, 7.24.3.1, 7.24.3.3

- error** preprocessing directive, 4, 6.10.5
- error-handling functions, 7.19.10, 7.21.6.2
- escape character (\), 6.4.4.4
- escape sequences, 5.2.1, 5.2.2, 6.4.4.4, 6.11.4
- evaluation order, 6.5
- exceptional condition, 6.5, 7.12.1
- exclusive OR operators
 - bitwise (^), 6.5.11
 - bitwise assignment (^=), 6.5.16.2
- executable program, 5.1.1.1
- execution character set, 5.2.1
- execution environment, 5, 5.1.2, *see also*
 - environmental limits
- execution sequence, 5.1.2.3, 6.8
- exit** function, 5.1.2.2.3, 7.19.3, 7.20, 7.20.4.3, 7.20.4.4
- EXIT_FAILURE** macro, 7.20, 7.20.4.3
- EXIT_SUCCESS** macro, 7.20, 7.20.4.3
- exp** functions, 7.12.6.1, F.9.3.1
- exp** type-generic macro, 7.22
- exp2** functions, 7.12.6.2, F.9.3.2
- exp2** type-generic macro, 7.22
- explicit conversion, 6.3
- expm1** functions, 7.12.6.3, F.9.3.3
- expm1** type-generic macro, 7.22
- exponent part, 6.4.4.2
- exponential functions
 - complex, 7.3.7, G.6.3
 - real, 7.12.6, F.9.3
- expression, 6.5
 - assignment, 6.5.16
 - cast, 6.5.4
 - constant, 6.6
 - full, 6.8
 - order of evaluation, 6.5
 - parenthesized, 6.5.1
 - primary, 6.5.1
 - unary, 6.5.3
- expression statement, 6.8.3
- extended character set, 3.7.2, 5.2.1, 5.2.1.2
- extended characters, 5.2.1
- extended integer types, 6.2.5, 6.3.1.1, 6.4.4.1, 7.18
- extended multibyte/wide character conversion utilities, 7.24.6
- extensible wide character case mapping functions, 7.25.3.2
- extensible wide character classification functions, 7.25.2.2
- extern** storage-class specifier, 6.2.2, 6.7.1
- external definition, 6.9
- external identifiers, underscore, 7.1.3
- external linkage, 6.2.2
- external name, 6.4.2.1
- external object definitions, 6.9.2
- fabs** functions, 7.12.7.2, F.9.4.2
- fabs** type-generic macro, 7.22, G.7
- false** macro, 7.16
- fclose** function, 7.19.5.1
- fdim** functions, 7.12.12.1, F.9.9.1
- fdim** type-generic macro, 7.22
- FE_ALL_EXCEPT** macro, 7.6
- FE_DFL_ENV** macro, 7.6
- FE_DIVBYZERO** macro, 7.6, 7.12, F.3
- FE_DOWNWARD** macro, 7.6, F.3
- FE_INEXACT** macro, 7.6, F.3
- FE_INVALID** macro, 7.6, 7.12, F.3
- FE_OVERFLOW** macro, 7.6, 7.12, F.3
- FE_TONEAREST** macro, 7.6, F.3
- FE_TOWARDZERO** macro, 7.6, F.3
- FE_UNDERFLOW** macro, 7.6, F.3
- FE_UPWARD** macro, 7.6, F.3
- feclearexcept** function, 7.6.2, 7.6.2.1, F.3
- fegetenv** function, 7.6.4.1, 7.6.4.3, 7.6.4.4, F.3
- fegetexceptflag** function, 7.6.2, 7.6.2.2, F.3
- fegetround** function, 7.6, 7.6.3.1, F.3
- feholdexcept** function, 7.6.4.2, 7.6.4.3, 7.6.4.4, F.3
- fenv.h** header, 5.1.2.3, 5.2.4.2.2, 7.6, 7.12, F, H
- FENV_ACCESS** pragma, 6.10.6, 7.6.1, F.7, F.8, F.9
- fenv_t** type, 7.6
- feof** function, 7.19.10.2
- feraiseexcept** function, 7.6.2, 7.6.2.3, F.3
- error** function, 7.19.10.3
- fesetenv** function, 7.6.4.3, F.3
- fesetexceptflag** function, 7.6.2, 7.6.2.4, F.3
- fesetround** function, 7.6, 7.6.3.2, F.3
- fetestexcept** function, 7.6.2, 7.6.2.5, F.3
- feupdateenv** function, 7.6.4.2, 7.6.4.4, F.3
- fexcept_t** type, 7.6, F.3
- fflush** function, 7.19.5.2, 7.19.5.3
- fgetc** function, 7.19.1, 7.19.3, 7.19.7.1, 7.19.7.5, 7.19.8.1
- fgetpos** function, 7.19.2, 7.19.9.1, 7.19.9.3
- fgets** function, 7.19.1, 7.19.7.2
- fgetwc** function, 7.19.1, 7.19.3, 7.24.3.1, 7.24.3.6
- fgetws** function, 7.19.1, 7.24.3.2
- field width, 7.19.6.1, 7.24.2.1
- file, 7.19.3

- access functions, **7.19.5**
- name, **7.19.3**
- operations, **7.19.4**
- position indicator, 7.19.1, 7.19.2, **7.19.3**,
7.19.5.3, 7.19.7.1, 7.19.7.3, 7.19.7.11,
7.19.8.1, 7.19.8.2, 7.19.9.1, 7.19.9.2,
7.19.9.3, 7.19.9.4, 7.19.9.5, 7.24.3.1,
7.24.3.3, 7.24.3.10
- positioning functions, **7.19.9**
- file scope, **6.2.1**, 6.9
- FILE** type, **7.19.1**, 7.19.3
- FILENAME_MAX** macro, **7.19.1**
- flags, **7.19.6.1**, **7.24.2.1**
 - floating-point status, *see* floating-point status flag
- flexible array member, **6.7.2.1**
- float** **_Complex** type, **6.2.5**
- float** **_Complex** type conversion, **6.3.1.6**,
6.3.1.7, **6.3.1.8**
- float** **_Imaginary** type, **G.2**
- float** type, **6.2.5**, 6.4.4.2, 6.7.2, F.2
- float** type conversion, **6.3.1.4**, **6.3.1.5**, **6.3.1.7**,
6.3.1.8
- float.h** header, 4, **5.2.4.2.2**, **7.7**, 7.20.1.3,
7.24.4.1.1
- float_t** type, **7.12**, J.5.6
- floating constant, **6.4.4.2**
- floating suffix, **f** or **F**, **6.4.4.2**
- floating type conversion, 6.3.1.4, 6.3.1.5, 6.3.1.7,
F.3, F.4
- floating types, **6.2.5**, 6.11.1
- floating-point accuracy, 5.2.4.2.2, 6.4.4.2, 6.5,
7.20.1.3, F.5, *see also* contracted expression
- floating-point arithmetic functions, **7.12**, **F.9**
- floating-point control mode, **7.6**, F.7.6
- floating-point environment, **7.6**, F.7, F.7.6
- floating-point exception, **7.6**, **7.6.2**, **F.9**
- floating-point number, **5.2.4.2.2**, **6.2.5**
- floating-point rounding mode, **5.2.4.2.2**
- floating-point status flag, **7.6**, F.7.6
- floor** functions, **7.12.9.2**, **F.9.6.2**
- floor** type-generic macro, **7.22**
- FLT_DIG** macro, **5.2.4.2.2**
- FLT_EPSILON** macro, **5.2.4.2.2**
- FLT_EVAL_METHOD** macro, **5.2.4.2.2**, 7.12
- FLT_MANT_DIG** macro, **5.2.4.2.2**
- FLT_MAX** macro, **5.2.4.2.2**
- FLT_MAX_10_EXP** macro, **5.2.4.2.2**
- FLT_MAX_EXP** macro, **5.2.4.2.2**
- FLT_MIN** macro, **5.2.4.2.2**
- FLT_MIN_10_EXP** macro, **5.2.4.2.2**
- FLT_MIN_EXP** macro, **5.2.4.2.2**
- FLT_RADIX** macro, **5.2.4.2.2**, 7.19.6.1, 7.20.1.3,
7.24.2.1, 7.24.4.1.1
- FLT_ROUNDS** macro, **5.2.4.2.2**, 7.6, 7.12.13.1,
F.3
- fma** functions, 7.12, **7.12.13.1**, **F.9.10.1**
- fma** type-generic macro, **7.22**
- fmax** functions, **7.12.12.2**, **F.9.9.2**
- fmax** type-generic macro, **7.22**
- fmin** functions, **7.12.12.3**, **F.9.9.3**
- fmin** type-generic macro, **7.22**
- fmod** functions, **7.12.10.1**, **F.9.7.1**
- fmod** type-generic macro, **7.22**
- fopen** function, **7.19.5.3**, 7.19.5.4
- FOPEN_MAX** macro, **7.19.1**, 7.19.3, 7.19.4.3
- for** statement, 6.8.5, **6.8.5.3**
- form-feed character, 5.2.1, **6.4**
- form-feed escape sequence (**\f**), **5.2.2**, 6.4.4.4,
7.4.1.10
- formal argument (deprecated), **3.15**
- formal parameter, **3.15**
- formatted input/output functions, 7.11.1.1, **7.19.6**
wide character, **7.24.2**
- fortran** keyword, **J.5.9**
- forward reference, **3.11**
- FP_CONTRACT** pragma, 6.5, 6.10.6, **7.12.2**, *see also* contracted expression
- FP_FAST_FMA** macro, **7.12**
- FP_FAST_FMAF** macro, **7.12**
- FP_FAST_FMAL** macro, **7.12**
- FP_ILOGB0** macro, **7.12**, 7.12.6.5
- FP_ILOGBNAN** macro, **7.12**, 7.12.6.5
- FP_INFINITE** macro, **7.12**, F.3
- FP_NAN** macro, **7.12**, F.3
- FP_NORMAL** macro, **7.12**, F.3
- FP_SUBNORMAL** macro, **7.12**, F.3
- FP_ZERO** macro, **7.12**, F.3
- fpclassify** macro, **7.12.3.1**, F.3
- fpos_t** type, **7.19.1**, 7.19.2
- fprintf** function, 7.8.1, 7.19.1, **7.19.6.1**,
7.19.6.2, 7.19.6.3, 7.19.6.5, 7.19.6.6,
7.19.6.8, 7.24.2.2, F.3
- fputc** function, 5.2.2, 7.19.1, 7.19.3, **7.19.7.3**,
7.19.7.8, 7.19.8.2
- fputs** function, 7.19.1, **7.19.7.4**
- fputwc** function, 7.19.1, 7.19.3, **7.24.3.3**,
7.24.3.8
- fputws** function, 7.19.1, **7.24.3.4**
- fread** function, 7.19.1, **7.19.8.1**
- free** function, **7.20.3.2**, 7.20.3.4
- freestanding execution environment, 4, 5.1.2,

- 5.1.2.1**
- freopen** function, 7.19.2, **7.19.5.4**
- frexp** functions, **7.12.6.4**, **F.9.3.4**
- frexp** type-generic macro, **7.22**
- fscanf** function, 7.8.1, 7.19.1, **7.19.6.2**,
7.19.6.4, 7.19.6.7, 7.19.6.9, F.3
- fseek** function, 7.19.1, 7.19.5.3, 7.19.7.11,
7.19.9.2, 7.19.9.4, 7.19.9.5, 7.24.3.10
- fsetpos** function, 7.19.2, 7.19.5.3, 7.19.7.11,
7.19.9.1, **7.19.9.3**, 7.24.3.10
- ftell** function, 7.19.9.2, **7.19.9.4**
- full declarator, **6.7.5**
- full expression, **6.8**
- fully buffered stream, **7.19.3**
- function
- argument, **6.5.2.2**, 6.9.1
 - body, **6.9.1**
 - call, **6.5.2.2**
 - library, **7.1.4**
 - declarator, **6.7.5.3**, 6.11.6
 - definition, 6.7.5.3, **6.9.1**, 6.11.7
 - designator, **6.3.2.1**
 - image, **5.2.3**
 - library, 5.1.1.1, **7.1.4**
 - name length, **5.2.4.1**, **6.4.2.1**, **6.11.3**
 - parameter, 5.1.2.2.1, **6.5.2.2**, 6.9.1
 - prototype, **6.2.1**, 6.5.2.2, **6.7.5.3**, 6.9.1
 - prototype scope, **6.2.1**
 - recursive call, **6.5.2.2**
 - return, **6.8.6.4**
 - scope, **6.2.1**
 - type, **6.2.5**
 - type conversion, **6.3.2.1**
- function specifiers, **6.7.4**
- function type, **6.2.5**
- function-call operator (()), **6.5.2.2**
- function-like macro, **6.10.3**
- future directions
- language, **6.11**
 - library, **7.26**
- fwide** function, 7.19.2, **7.24.3.5**
- fwprintf** function, 7.8.1, 7.19.1, 7.19.6.2,
7.24.2.1, 7.24.2.2, 7.24.2.3, 7.24.2.5,
7.24.2.11
- fwrite** function, 7.19.1, **7.19.8.2**
- fwscanf** function, 7.8.1, 7.19.1, **7.24.2.2**,
7.24.2.4, 7.24.2.6, 7.24.2.12, 7.24.3.10
- gamma functions, **7.12.8**, **F.9.5**
- general utilities, **7.20**
- wide string, **7.24.4**
- general wide string utilities, **7.24.4**
- generic parameters, **7.22**
- getc** function, 7.19.1, **7.19.7.5**, 7.19.7.6
- getchar** function, 7.19.1, **7.19.7.6**
- getenv** function, **7.20.4.5**
- gets** function, 7.19.1, **7.19.7.7**
- getwc** function, 7.19.1, **7.24.3.6**, 7.24.3.7
- getwchar** function, 7.19.1, **7.24.3.7**
- gmtime** function, **7.23.3.3**
- goto** statement, 6.2.1, 6.8.1, **6.8.6.1**
- graphic characters, **5.2.1**
- greater-than operator (>), **6.5.8**
- greater-than-or-equal-to operator (>=), **6.5.8**
- header, 5.1.1.1, **7.1.2**, *see also* standard headers
- header names, 6.4, **6.4.7**, 6.10.2
- hexadecimal constant, **6.4.4.1**
- hexadecimal digit, **6.4.4.1**, 6.4.4.2, 6.4.4.4
- hexadecimal prefix, **6.4.4.1**
- hexadecimal-character escape sequence
(\x *hexadecimal digits*), **6.4.4.4**
- high-order bit, **3.6**
- horizontal-tab character, 5.2.1, **6.4**
- horizontal-tab escape sequence (\r), 7.25.2.1.3
- horizontal-tab escape sequence (\t), **5.2.2** ,
6.4.4.4, 7.4.1.3, 7.4.1.10
- hosted execution environment, **4**, 5.1.2, **5.1.2.2**
- HUGE_VAL** macro, **7.12**, 7.12.1, 7.20.1.3,
7.24.4.1.1, F.9
- HUGE_VALF** macro, **7.12**, 7.12.1, 7.20.1.3,
7.24.4.1.1, F.9
- HUGE_VALL** macro, **7.12**, 7.12.1, 7.20.1.3,
7.24.4.1.1, F.9
- hyperbolic functions
- complex, **7.3.6**, **G.6.2**
 - real, **7.12.5**, **F.9.2**
- hypot** functions, **7.12.7.3**, **F.9.4.3**
- hypot** type-generic macro, **7.22**
- I** macro, **7.3.1**, 7.3.9.4, **G.6**
- identifier, **6.4.2.1**, 6.5.1
- linkage, *see* linkage
 - maximum length, **6.4.2.1**
 - name spaces, **6.2.3**
 - reserved, **6.4.1**, **7.1.3**
 - scope, **6.2.1**
 - type, **6.2.5**
- identifier list, **6.7.5**
- identifier nondigit, **6.4.2.1**
- IEC 559, **F.1**
- IEC 60559, 2, 5.1.2.3, **5.2.4.2.2**, 6.10.8, 7.3.3, 7.6,

- 7.6.4.2, 7.12.1, 7.12.10.2, 7.12.14, **F**, **G**, H.1
- IEEE 754, **F.1**
- IEEE 854, **F.1**
- IEEE floating-point arithmetic standard, *see*
 - IEC 60559, ANSI/IEEE 754,
 - ANSI/IEEE 854
- if** preprocessing directive, 5.2.4.2.1, 5.2.4.2.2, **6.10.1**, 7.1.4
- if** statement, **6.8.4.1**
- ifdef** preprocessing directive, **6.10.1**
- ifndef** preprocessing directive, **6.10.1**
- ilogb** functions, 7.12, **7.12.6.5**, **F.9.3.5**
- ilogb** type-generic macro, **7.22**
- imaginary** macro, **7.3.1**, **G.6**
- imaginary numbers, **G**
- imaginary type domain, **G.2**
- imaginary types, 6.7.2, **G**
- imaxabs** function, **7.8.2.1**
- imaxdiv** function, 7.8, **7.8.2.2**
- imaxdiv_t** type, **7.8**
- implementation, **3.12**
- implementation limit, **3.13**, 4, 5.2.4.2, 6.4.2.1, 6.7.5, 6.8.4.2, E, *see also* environmental limits
- implementation-defined behavior, **3.4.1**, 4, J.3
- implementation-defined value, **3.17.1**
- implicit conversion, **6.3**
- implicit initialization, **6.7.8**
- include** preprocessing directive, 5.1.1.2, **6.10.2**
- inclusive OR operators
 - bitwise (**|**), **6.5.12**
 - bitwise assignment (**|=**), **6.5.16.2**
- incomplete type, **6.2.5**
- increment operators, *see* arithmetic operators,
 - increment and decrement
- indeterminate value, **3.17.2**
- indirection operator (*****), 6.5.2.1, **6.5.3.2**
- inequality operator (**!=**), **6.5.9**
- INFINITY** macro, 7.3.9.4, **7.12**, F.2.1
- initial position, 5.2.2
- initial shift state, **5.2.1.2**
- initialization, 5.1.2, 6.2.4, 6.3.2.1, 6.5.2.5, **6.7.8**, F.7.5
 - in blocks, 6.8
- initializer, **6.7.8**
 - permitted form, **6.6**
 - string literal, 6.3.2.1
- inline, **6.7.4**
- inner scope, **6.2.1**
- input failure, 7.24.2.6, 7.24.2.8, 7.24.2.10
- input/output functions
 - character, **7.19.7**
 - direct, **7.19.8**
 - formatted, **7.19.6**
 - wide character, **7.24.2**
 - wide character, **7.24.3**
 - formatted, **7.24.2**
- input/output header, **7.19**
- input/output, device, **5.1.2.3**
- int** type, **6.2.5**, 6.3.1.1, 6.3.1.3, 6.4.4.1, 6.7.2
- int** type conversion, **6.3.1.1**, **6.3.1.3**, **6.3.1.4**, **6.3.1.8**
- INT_FASTN_MAX** macros, **7.18.2.3**
- INT_FASTN_MIN** macros, **7.18.2.3**
- int_fastN_t** types, **7.18.1.3**
- INT_LEASTN_MAX** macros, **7.18.2.2**
- INT_LEASTN_MIN** macros, **7.18.2.2**
- int_leastN_t** types, **7.18.1.2**
- INT_MAX** macro, **5.2.4.2.1**, 7.12, 7.12.6.5
- INT_MIN** macro, **5.2.4.2.1**, 7.12
- integer arithmetic functions, **7.8.2.1**, **7.8.2.2**, **7.20.6**
- integer character constant, **6.4.4.4**
- integer constant, **6.4.4.1**
- integer constant expression, **6.6**
- integer conversion rank, **6.3.1.1**
- integer promotions, 5.1.2.3, 5.2.4.2.1, **6.3.1.1**, 6.5.2.2, 6.5.3.3, 6.5.7, 6.8.4.2, 7.18.2, 7.18.3, 7.19.6.1, 7.24.2.1
- integer suffix, **6.4.4.1**
- integer type conversion, **6.3.1.1**, 6.3.1.3, 6.3.1.4, F.3, F.4
- integer types, **6.2.5**, 7.18
 - extended, **6.2.5**, 6.3.1.1, 6.4.4.1, **7.18**
- interactive device, **5.1.2.3**, 7.19.3, 7.19.5.3
- internal linkage, **6.2.2**
- internal name, **6.4.2.1**
- interrupt, **5.2.3**
- INTMAX_C** macro, **7.18.4.2**
- INTMAX_MAX** macro, 7.8.2.3, 7.8.2.4, **7.18.2.5**
- INTMAX_MIN** macro, 7.8.2.3, 7.8.2.4, **7.18.2.5**
- intmax_t** type, **7.18.1.5**, 7.19.6.1, 7.19.6.2, 7.24.2.1, 7.24.2.2
- INTN_C** macros, **7.18.4.1**
- INTN_MAX** macros, **7.18.2.1**
- INTN_MIN** macros, **7.18.2.1**
- intN_t** types, **7.18.1.1**
- INTPTR_MAX** macro, **7.18.2.4**
- INTPTR_MIN** macro, **7.18.2.4**
- intptr_t** type, **7.18.1.4**
- inttypes.h** header, **7.8**, 7.26.4
- isalnum** function, **7.4.1.1**, 7.4.1.9, 7.4.1.10

- isalpha** function, 7.4.1.1, **7.4.1.2**
- isblank** function, **7.4.1.3**
- iscntrl** function, 7.4.1.2, **7.4.1.4**, 7.4.1.7, 7.4.1.11
- isdigit** function, 7.4.1.1, 7.4.1.2, **7.4.1.5**, 7.4.1.7, 7.4.1.11, 7.11.1.1
- isfinite** macro, **7.12.3.2**, F.3
- isgraph** function, **7.4.1.6**
- isgreater** macro, **7.12.14.1**, F.3
- isgreaterequal** macro, **7.12.14.2**, F.3
- isinf** macro, **7.12.3.3**
- isless** macro, **7.12.14.3**, F.3
- islessequal** macro, **7.12.14.4**, F.3
- islessgreater** macro, **7.12.14.5**, F.3
- islower** function, 7.4.1.2, **7.4.1.7**, 7.4.2.1, 7.4.2.2
- isnan** macro, **7.12.3.4**, F.3
- isnormal** macro, **7.12.3.5**
- ISO 31–11, 2, 3
- ISO 4217, 2, 7.11.2.1
- ISO 8601, 2, **7.23.3.5**
- ISO/IEC 10646, 2, 6.4.2.1, **6.4.3**, 6.10.8
- ISO/IEC 10976–1, H.1
- ISO/IEC 2382–1, 2, 3
- ISO/IEC 646, 2, **5.2.1.1**
- ISO/IEC 9945–2, 7.11
- ISO/IEC TR 10176, **D**
- iso646.h** header, 4, **7.9**
- isprint** function, 5.2.2, **7.4.1.8**
- ispunct** function, 7.4.1.2, 7.4.1.7, **7.4.1.9**, 7.4.1.11
- isspace** function, 7.4.1.2, 7.4.1.7, 7.4.1.9, **7.4.1.10**, 7.4.1.11, 7.19.6.2, 7.20.1.3, 7.20.1.4, 7.24.2.2
- isunordered** macro, **7.12.14.6**, F.3
- isupper** function, 7.4.1.2, **7.4.1.11**, 7.4.2.1, 7.4.2.2
- iswalnum** function, **7.25.2.1.1**, 7.25.2.1.9, 7.25.2.1.10, 7.25.2.2.1
- iswalpha** function, 7.25.2.1.1, **7.25.2.1.2**, 7.25.2.2.1
- iswblank** function, **7.25.2.1.3**, 7.25.2.2.1
- iswcntrl** function, 7.25.2.1.2, **7.25.2.1.4**, 7.25.2.1.7, 7.25.2.1.11, 7.25.2.2.1
- iswctype** function, **7.25.2.2.1**, 7.25.2.2.2
- iswdigit** function, 7.25.2.1.1, 7.25.2.1.2, **7.25.2.1.5**, 7.25.2.1.7, 7.25.2.1.11, 7.25.2.2.1
- iswgraph** function, 7.25.2.1, **7.25.2.1.6**, 7.25.2.1.10, 7.25.2.2.1
- iswlower** function, 7.25.2.1.2, **7.25.2.1.7**, 7.25.2.2.1, 7.25.3.1.1, 7.25.3.1.2
- iswprint** function, 7.25.2.1.6, **7.25.2.1.8**, 7.25.2.2.1
- iswpunct** function, 7.25.2.1, 7.25.2.1.2, 7.25.2.1.7, **7.25.2.1.9**, 7.25.2.1.10, 7.25.2.1.11, 7.25.2.2.1
- iswspace** function, 7.19.6.2, 7.24.2.2, 7.24.4.1.1, 7.24.4.1.2, 7.25.2.1.2, 7.25.2.1.6, 7.25.2.1.7, 7.25.2.1.9, **7.25.2.1.10**, 7.25.2.1.11, 7.25.2.2.1
- iswupper** function, 7.25.2.1.2, **7.25.2.1.11**, 7.25.2.2.1, 7.25.3.1.1, 7.25.3.1.2
- iswxdigit** function, **7.25.2.1.12**, 7.25.2.2.1
- isxdigit** function, **7.4.1.12**, 7.11.1.1
- italic type* convention, 3, **6.1**
- iteration statements, **6.8.5**
- jmp_buf** type, **7.13**
- jump statements, **6.8.6**
- keywords, **6.4.1**
- L_tmpnam** macro, **7.19.1**, 7.19.4.4
- label name, 6.2.1, **6.2.3**
- labeled statement, **6.8.1**
- labs** function, **7.20.6.1**
- language, **6**
 - future directions, **6.11**
 - syntax summary, **A**
- Latin alphabet, **5.2.1**, 6.4.2.1
- LC_ALL** macro, **7.11**, **7.11.1.1**, 7.11.2.1
- LC_COLLATE** macro, **7.11**, **7.11.1.1**, 7.21.4.3, 7.24.4.4.2
- LC_CTYPE** macro, **7.11**, **7.11.1.1**, 7.20, 7.20.7, 7.20.8, 7.24.6, 7.25.1, 7.25.2.2.1, 7.25.2.2.2, 7.25.3.2.1, 7.25.3.2.2
- LC_MONETARY** macro, **7.11**, **7.11.1.1**, 7.11.2.1
- LC_NUMERIC** macro, **7.11**, **7.11.1.1**, 7.11.2.1
- LC_TIME** macro, **7.11**, **7.11.1.1**, 7.23.3.5
- lconv** structure type, **7.11**
- LDBL_DIG** macro, **5.2.4.2.2**
- LDBL_EPSILON** macro, **5.2.4.2.2**
- LDBL_MANT_DIG** macro, **5.2.4.2.2**
- LDBL_MAX** macro, **5.2.4.2.2**
- LDBL_MAX_10_EXP** macro, **5.2.4.2.2**
- LDBL_MAX_EXP** macro, **5.2.4.2.2**
- LDBL_MIN** macro, **5.2.4.2.2**
- LDBL_MIN_10_EXP** macro, **5.2.4.2.2**
- LDBL_MIN_EXP** macro, **5.2.4.2.2**
- ldexp** functions, **7.12.6.6**, **F.9.3.6**
- ldexp** type-generic macro, **7.22**
- ldiv** function, **7.20.6.2**

- ldiv_t** type, **7.20**
- leading underscore in identifiers, **7.1.3**
- left-shift assignment operator (<<=), **6.5.16.2**
- left-shift operator (<<), **6.5.7**
- length
 - external name, **5.2.4.1, 6.4.2.1, 6.11.3**
 - function name, **5.2.4.1, 6.4.2.1, 6.11.3**
 - identifier, **6.4.2.1**
 - internal name, **5.2.4.1, 6.4.2.1**
- length function, **7.20.7.1, 7.21.6.3, 7.24.4.6.1, 7.24.6.3.1**
- length modifier, **7.19.6.1, 7.19.6.2, 7.24.2.1, 7.24.2.2**
- less-than operator (<), **6.5.8**
- less-than-or-equal-to operator (<=), **6.5.8**
- letter, **5.2.1, 7.4**
- lexical elements, **5.1.1.2, 6.4**
- lgamma** functions, **7.12.8.3, F.9.5.3**
- lgamma** type-generic macro, **7.22**
- library, **5.1.1.1, 7**
 - future directions, **7.26**
 - summary, **B**
 - terms, **7.1.1**
 - use of functions, **7.1.4**
- lifetime, **6.2.4**
- limits
 - environmental, *see* environmental limits
 - implementation, *see* implementation limits
 - numerical, *see* numerical limits
 - translation, *see* translation limits
- limits.h** header, **4, 5.2.4.2.1, 6.2.5, 7.10**
- line buffered stream, **7.19.3**
- line number, **6.10.4, 6.10.8**
- line** preprocessing directive, **6.10.4**
- lines, **5.1.1.2, 7.19.2**
 - preprocessing directive, **6.10**
- linkage, **6.2.2, 6.7, 6.7.4, 6.7.5.2, 6.9, 6.9.2, 6.11.2**
- labs** function, **7.20.6.1**
- lldiv** function, **7.20.6.2**
- lldiv_t** type, **7.20**
- LLONG_MAX** macro, **5.2.4.2.1, 7.20.1.4, 7.24.4.1.2**
- LLONG_MIN** macro, **5.2.4.2.1, 7.20.1.4, 7.24.4.1.2**
- llrint** functions, **7.12.9.5, F.3, F.9.6.5**
- llrint** type-generic macro, **7.22**
- llround** functions, **7.12.9.7, F.9.6.7**
- llround** type-generic macro, **7.22**
- local time, **7.23.1**
- locale, **3.4.2**
 - locale-specific behavior, **3.4.2, J.4**
 - locale.h** header, **7.11, 7.26.5**
 - localeconv** function, **7.11.1.1, 7.11.2.1**
 - localization, **7.11**
 - localtime** function, **7.23.3.4**
 - log** functions, **7.12.6.7, F.9.3.7**
 - log** type-generic macro, **7.22**
 - log10** functions, **7.12.6.8, F.9.3.8**
 - log10** type-generic macro, **7.22**
 - log1p** functions, **7.12.6.9, F.9.3.9**
 - log1p** type-generic macro, **7.22**
 - log2** functions, **7.12.6.10, F.9.3.10**
 - log2** type-generic macro, **7.22**
 - logarithmic functions
 - complex, **7.3.7, G.6.3**
 - real, **7.12.6, F.9.3**
 - logb** functions, **7.12.6.11, F.3, F.9.3.11**
 - logb** type-generic macro, **7.22**
 - logical operators
 - AND (&&), **6.5.13**
 - negation (!), **6.5.3.3**
 - OR (| |), **6.5.14**
 - logical source lines, **5.1.1.2**
 - long double _Complex** type, **6.2.5**
 - long double _Complex** type conversion, **6.3.1.6, 6.3.1.7, 6.3.1.8**
 - long double _Imaginary** type, **G.2**
 - long double suffix, **l** or **L**, **6.4.4.2**
 - long double** type, **6.2.5, 6.4.4.2, 6.7.2, 7.19.6.1, 7.19.6.2, 7.24.2.1, 7.24.2.2, F.2**
 - long double** type conversion, **6.3.1.4, 6.3.1.5, 6.3.1.7, 6.3.1.8**
 - long int** type, **6.2.5, 6.3.1.1, 6.7.2, 7.19.6.1, 7.19.6.2, 7.24.2.1, 7.24.2.2**
 - long int** type conversion, **6.3.1.1, 6.3.1.3, 6.3.1.4, 6.3.1.8**
 - long integer suffix, **l** or **L**, **6.4.4.1**
 - long long int** type, **6.2.5, 6.3.1.1, 6.7.2, 7.19.6.1, 7.19.6.2, 7.24.2.1, 7.24.2.2**
 - long long int** type conversion, **6.3.1.1, 6.3.1.3, 6.3.1.4, 6.3.1.8**
 - long long integer suffix, **ll** or **LL**, **6.4.4.1**
 - LONG_MAX** macro, **5.2.4.2.1, 7.20.1.4, 7.24.4.1.2**
 - LONG_MIN** macro, **5.2.4.2.1, 7.20.1.4, 7.24.4.1.2**
 - longjmp** function, **7.13.1.1, 7.13.2.1, 7.20.4.3**
 - loop body, **6.8.5**
 - low-order bit, **3.6**
 - lowercase letter, **5.2.1**
 - lrint** functions, **7.12.9.5, F.3, F.9.6.5**
 - lrint** type-generic macro, **7.22**
 - lround** functions, **7.12.9.7, F.9.6.7**

- lround** type-generic macro, **7.22**
- lvalue**, **6.3.2.1**, 6.5.1, 6.5.2.4, 6.5.3.1, 6.5.16
- macro argument substitution, **6.10.3.1**
- macro definition
 - library function, **7.1.4**
- macro invocation, 6.10.3
- macro name, **6.10.3**
 - length, **5.2.4.1**
 - predefined, **6.10.8**, 6.11.9
 - redefinition, **6.10.3**
 - scope, **6.10.3.5**
- macro parameter, **6.10.3**
- macro preprocessor, **6.10**
- macro replacement, **6.10.3**
- magnitude, complex, 7.3.8.1
- main** function, **5.1.2.2.1**, 5.1.2.2.3, 6.7.3.1, 6.7.4, 7.19.3
- malloc** function, 7.20.3, 7.20.3.2, **7.20.3.3**, 7.20.3.4
- manipulation functions
 - complex, **7.3.9**
 - real, **7.12.11**, **F.9.8**
- matching failure, 7.24.2.6, 7.24.2.8, 7.24.2.10
- math.h** header, 5.2.4.2.2, 6.5, **7.12**, 7.22, **F**, **F.9**, J.5.17
- MATH_ERREXCEPT** macro, **7.12**, **F.9**
- math_errhandling** macro, 7.1.3, **7.12**, **F.9**
- MATH_ERRNO** macro, **7.12**
- maximum functions, **7.12.12**, **F.9.9**
- MB_CUR_MAX** macro, 7.1.1, **7.20**, 7.20.7.2, 7.20.7.3, 7.24.6.3.3
- MB_LEN_MAX** macro, **5.2.4.2.1**, 7.1.1, 7.20
- mblen** function, **7.20.7.1**, 7.24.6.3
- mbrlen** function, **7.24.6.3.1**
- mbrtowc** function, 7.19.3, 7.19.6.1, 7.19.6.2, 7.24.2.1, 7.24.2.2, 7.24.6.3.1, **7.24.6.3.2**, 7.24.6.4.1
- mbsinit** function, **7.24.6.2.1**
- mbsrtowcs** function, **7.24.6.4.1**
- mbstate_t** type, 7.19.2, 7.19.3, 7.19.6.1, 7.19.6.2, **7.24.1**, 7.24.2.1, 7.24.2.2, 7.24.6, 7.24.6.2.1, 7.24.6.3, 7.24.6.3.1, 7.24.6.4
- mbstowcs** function, 6.4.5, **7.20.8.1**, 7.24.6.4
- mbtowc** function, 7.20.7.1, **7.20.7.2**, 7.20.8.1, 7.24.6.3
- member access operators (**.** and **->**), **6.5.2.3**
- member alignment, **6.7.2.1**
- memchr** function, **7.21.5.1**
- memcmp** function, 7.21.4, **7.21.4.1**
- memcpy** function, **7.21.2.1**
- memmove** function, **7.21.2.2**
- memory management functions, **7.20.3**
- memset** function, **7.21.6.1**
- minimum functions, **7.12.12**, **F.9.9**
- minus operator, unary, **6.5.3.3**
- miscellaneous functions
 - string, **7.21.6**
 - wide string, **7.24.4.6**
- mktime** function, **7.23.2.3**
- modf** functions, **7.12.6.12**, **F.9.3.12**
- modifiable **lvalue**, **6.3.2.1**
- modulus functions, **7.12.6.12**
- modulus, complex, 7.3.8.1
- multibyte character, **3.7.2**, **5.2.1.2**, 6.4.4.4
- multibyte conversion functions
 - wide character, **7.20.7**
 - extended, **7.24.6**
 - restartable, **7.24.6.3**
 - wide string, **7.20.8**
 - restartable, **7.24.6.4**
- multibyte string, **7.1.1**
- multibyte/wide character conversion functions, **7.20.7**
 - extended, **7.24.6**
 - restartable, **7.24.6.3**
- multibyte/wide string conversion functions, **7.20.8**
 - restartable, **7.24.6.4**
- multidimensional array, 6.5.2.1
- multiplication assignment operator (***=**), **6.5.16.2**
- multiplication operator (*****), **6.5.5**, **F.3**, **G.5.1**
- multiplicative expressions, **6.5.5**, **G.5.1**
- n-char sequence, **7.20.1.3**
- n-wchar sequence, **7.24.4.1.1**
- name
 - external, 5.2.4.1, **6.4.2.1**, **6.11.3**
 - file, **7.19.3**
 - internal, 5.2.4.1, **6.4.2.1**
 - label, **6.2.3**
 - structure/union member, **6.2.3**
- name spaces, **6.2.3**
- named label, **6.8.1**
- NaN, **5.2.4.2.2**
- nan** functions, **7.12.11.2**, **F.2.1**, **F.9.8.2**
- NAN** macro, **7.12**, **F.2.1**
- NDEBUG** macro, **7.2**
- nearbyint** functions, **7.12.9.3**, 7.12.9.4, **F.3**, **F.9.6.3**
- nearbyint** type-generic macro, **7.22**
- nearest integer functions, **7.12.9**, **F.9.6**
- negation operator (**!**), **6.5.3.3**

- negative zero, **6.2.6.2**
- new-line character, **5.1.1.2**, **5.2.1**, 6.4, 6.10, 6.10.4
- new-line escape sequence (`\n`), **5.2.2**, 6.4.4.4, 7.4.1.10
- nextafter** functions, **7.12.11.3**, 7.12.11.4, F.3, **F.9.8.3**
- nextafter** type-generic macro, **7.22**
- nexttoward** functions, **7.12.11.4**, F.3, **F.9.8.4**
- nexttoward** type-generic macro, **7.22**
- no linkage, **6.2.2**
- non-stop floating-point control mode, **7.6.4.2**
- nongraphic characters, **5.2.2**, 6.4.4.4
- nonlocal jumps header, **7.13**
- norm, complex, 7.3.8.1
- not** macro, **7.9**
- not-equal-to operator, *see* inequality operator
- not_eq** macro, **7.9**
- null character (`\0`), **5.2.1**, 6.4.4.4, 6.4.5
 - padding of binary stream, **7.19.2**
- NULL** macro, 7.11, **7.17**, **7.19.1**, **7.20**, **7.21.1**, **7.23.1**, **7.24.1**
- null pointer, **6.3.2.3**
- null pointer constant, **6.3.2.3**
- null preprocessing directive, **6.10.7**
- null statement, **6.8.3**
- null wide character, **7.1.1**
- numeric conversion functions, **7.8.2.3**, **7.20.1**
 - wide string, **7.8.2.4**, **7.24.4.1**
- numerical limits, **5.2.4.2**
- object, **3.14**
- object representation, **6.2.6.1**
- object type, **6.2.5**
- object-like macro, **6.10.3**
- obsolescence, **6.11**, **7.26**
- octal constant, **6.4.4.1**
- octal digit, **6.4.4.1**, 6.4.4.4
- octal-character escape sequence (*octal digits*), **6.4.4.4**
- offsetof** macro, **7.17**
- on-off switch, **6.10.6**
- one's complement, **6.2.6.2**
- operand, **6.4.6**, 6.5
- operating system, **5.1.2.1**, 7.20.4.6
- operations on files, 7.19.4
- operator, **6.4.6**
- operators, 6.5
 - assignment, **6.5.16**
 - associativity, **6.5**
 - equality, **6.5.9**
 - multiplicative, **6.5.5**, G.5.1
 - postfix, **6.5.2**
 - precedence, **6.5**
 - preprocessing, **6.10.1**, **6.10.3.2**, **6.10.3.3**, **6.10.9**
 - relational, **6.5.8**
 - shift, **6.5.7**
 - unary, **6.5.3**
 - unary arithmetic, **6.5.3.3**
- or** macro, **7.9**
- OR operators
 - bitwise exclusive (`^`), **6.5.11**
 - bitwise exclusive assignment (`^=`), **6.5.16.2**
 - bitwise inclusive (`|`), **6.5.12**
 - bitwise inclusive assignment (`|=`), **6.5.16.2**
 - logical (`||`), **6.5.14**
- or_eq** macro, **7.9**
- order of allocated storage, **7.20.3**
- order of evaluation, **6.5**
- ordinary identifier name space, **6.2.3**
- orientation of stream, **7.19.2**, 7.24.3.5
- outer scope, **6.2.1**
- padding
 - binary stream, **7.19.2**
 - structure/union, **6.7.2.1**
- padding bits, **6.2.6.2**, 7.18.1.1
- parameter, **3.15**
 - array, **6.9.1**
 - ellipsis, **6.7.5.3**, **6.10.3**
 - function, **6.5.2.2**, 6.9.1
 - macro, **6.10.3**
 - main** function, **5.1.2.2.1**
 - program, **5.1.2.2.1**
- parameter type list, **6.7.5.3**
- parentheses punctuator (`()`), 6.7.5.3, 6.8.4, 6.8.5
- parenthesized expression, **6.5.1**
- parse state, 7.19.2
- permitted form of initializer, **6.6**
- perror** function, **7.19.10.4**
- phase angle, complex, 7.3.9.1
- physical source lines, **5.1.1.2**
- placemaker, **6.10.3.3**
- plus operator, unary, **6.5.3.3**
- pointer arithmetic, **6.5.6**
- pointer comparison, **6.5.8**
- pointer declarator, **6.7.5.1**
- pointer operator (`->`), **6.5.2.3**
- pointer to function, **6.5.2.2**
- pointer type, **6.2.5**
- pointer type conversion, 6.3.2.1, **6.3.2.3**
- pointer, null, **6.3.2.3**
- portability, 4, **J**

- position indicator, file, *see* file position indicator
- positive difference, **7.12.12.1**
- positive difference functions, **7.12.12, F.9.9**
- postfix decrement operator (`--`), 6.3.2.1, **6.5.2.4**
- postfix expressions, **6.5.2**
- postfix increment operator (`++`), 6.3.2.1, **6.5.2.4**
- pow** functions, **7.12.7.4, F.9.4.4**
- pow** type-generic macro, **7.22**
- power functions
 - complex, **7.3.8, G.6.4**
 - real, **7.12.7, F.9.4**
- pp-number, 6.4.8
- pragma operator, **6.10.9**
- pragma** preprocessing directive, **6.10.6**, 6.11.8
- precedence of operators, **6.5**
- precedence of syntax rules, **5.1.1.2**
- precision, **6.2.6.2**, 6.3.1.1, 6.3.1.8, **7.19.6.1, 7.24.2.1**
- predefined macro names, **6.10.8**, 6.11.9
- prefix decrement operator (`--`), 6.3.2.1, **6.5.3.1**
- prefix increment operator (`++`), 6.3.2.1, **6.5.3.1**
- preprocessing concatenation, **6.10.3.3**
- preprocessing directives, 5.1.1.2, **6.10**
- preprocessing file, **5.1.1.1, 6.10**
- preprocessing numbers, 6.4, **6.4.8**
- preprocessing operators
 - #**, **6.10.3.2**
 - ##**, **6.10.3.3**
 - _Pragma**, 5.1.1.2, **6.10.9**
 - defined**, **6.10.1**
- preprocessing tokens, 5.1.1.2, **6.4**, 6.10
- preprocessing translation unit, **5.1.1.1**
- preprocessor, **6.10**
- PRIcFASTN** macros, **7.8.1**
- PRIcLEASTN** macros, **7.8.1**
- PRIcMAX** macros, **7.8.1**
- PRIcN** macros, **7.8.1**
- PRIcPTR** macros, **7.8.1**
- primary expression, **6.5.1**
- printf** function, 7.19.1, **7.19.6.3**, 7.19.6.10
- printing character, 5.2.2, **7.4**, 7.4.1.8
- printing wide character, **7.25.2**
- program diagnostics, **7.2.1**
- program execution, 5.1.2.2.2, **5.1.2.3**
- program file, **5.1.1.1**
- program image, **5.1.1.2**
- program name (`argv[0]`), **5.1.2.2.1**
- program parameters, **5.1.2.2.1**
- program startup, **5.1.2**, 5.1.2.1, 5.1.2.2.1
- program structure, **5.1.1.1**
- program termination, **5.1.2**, 5.1.2.1, 5.1.2.2.3, 5.1.2.3
- program, conforming, **4**
- program, strictly conforming, **4**
- promotions
 - default argument, **6.5.2.2**
 - integer, 5.1.2.3, **6.3.1.1**
- prototype, *see* function prototype
- pseudo-random sequence functions, **7.20.2**
- PTRDIFF_MAX** macro, **7.18.3**
- PTRDIFF_MIN** macro, **7.18.3**
- ptrdiff_t** type, **7.17**, 7.18.3, 7.19.6.1, 7.19.6.2, 7.24.2.1, 7.24.2.2
- punctuators, **6.4.6**
- putc** function, 7.19.1, **7.19.7.8**, 7.19.7.9
- putchar** function, 7.19.1, **7.19.7.9**
- puts** function, 7.19.1, **7.19.7.10**
- putwc** function, 7.19.1, **7.24.3.8**, 7.24.3.9
- putwchar** function, 7.19.1, **7.24.3.9**
- qsort** function, 7.20.5, **7.20.5.2**
- qualified types, **6.2.5**
- qualified version of type, **6.2.5**
- question-mark escape sequence (`\?`), **6.4.4.4**
- quiet NaN, **5.2.4.2.2**
- raise** function, 7.14, 7.14.1.1, **7.14.2.1**, 7.20.4.1
- rand** function, 7.20, **7.20.2.1**, 7.20.2.2
- RAND_MAX** macro, **7.20**, 7.20.2.1
- range error, **7.12.1**, 7.12.5.3, 7.12.5.4, 7.12.5.5, 7.12.6.1, 7.12.6.2, 7.12.6.3, 7.12.6.5, 7.12.6.6, 7.12.6.7, 7.12.6.8, 7.12.6.9, 7.12.6.10, 7.12.6.13, 7.12.7.3, 7.12.7.4, 7.12.8.2, 7.12.8.3, 7.12.8.4, 7.12.9.5, 7.12.11.3, 7.12.12.1
- rank, *see* integer conversion rank
- real floating type conversion, 6.3.1.4, 6.3.1.5, 6.3.1.7, F.3, F.4
- real floating types, **6.2.5**
- real type domain, **6.2.5**
- real types, **6.2.5**
- real-floating, **7.12.3**
- realloc** function, 7.20.3, 7.20.3.2, **7.20.3.4**
- recommended practice, **3.16**
- recursion, 6.5.2.2
- recursive function call, **6.5.2.2**
- redefinition of macro, **6.10.3**
- reentrancy, 5.1.2.3, **5.2.3**
 - library functions, **7.1.4**
- referenced type, **6.2.5**
- register** storage-class specifier, **6.7.1**, 6.9
- relational expressions, **6.5.8**

- reliability of data, interrupted, **5.1.2.3**
- remainder assignment operator (**%=**), **6.5.16.2**
- remainder functions, **7.12.10**, **F.9.7**
- remainder** functions, **7.12.10.2**, 7.12.10.3, F.3, **F.9.7.2**
- remainder operator (**%**), **6.5.5**
- remainder** type-generic macro, **7.22**
- remove** function, **7.19.4.1**, 7.19.4.4
- remquo** functions, **7.12.10.3**, F.3, **F.9.7.3**
- remquo** type-generic macro, **7.22**
- rename** function, **7.19.4.2**
- representations of types, **6.2.6**
 - pointer, **6.2.5**
- rescanning and replacement, 6.10.3.4
- reserved identifiers, **6.4.1**, **7.1.3**
- restartable multibyte/wide character conversion functions, **7.24.6.3**
- restartable multibyte/wide string conversion functions, **7.24.6.4**
- restore calling environment function, **7.13.2**
- restrict** type qualifier, **6.7.3**, **6.7.3.1**
- restrict-qualified type, **6.2.5**, 6.7.3
- return** statement, **6.8.6.4**
- rewind** function, 7.19.5.3, 7.19.7.11, **7.19.9.5**, 7.24.3.10
- right-shift assignment operator (**>>=**), **6.5.16.2**
- right-shift operator (**>>**), **6.5.7**
- rint** functions, **7.12.9.4**, F.3, **F.9.6.4**
- rint** type-generic macro, **7.22**
- round** functions, **7.12.9.6**, **F.9.6.6**
- round** type-generic macro, **7.22**
- rounding mode, floating point, 5.2.4.2.2
- rvalue, **6.3.2.1**
- same scope, **6.2.1**
- save calling environment function, **7.13.1**
- scalar types, **6.2.5**
- scalbln** function, **7.12.6.13**, F.3, **F.9.3.13**
- scalbln** type-generic macro, **7.22**
- scalbn** function, **7.12.6.13**, F.3, **F.9.3.13**
- scalbn** type-generic macro, **7.22**
- scanf** function, 7.19.1, **7.19.6.4**, 7.19.6.11
- scanlist, **7.19.6.2**, **7.24.2.2**
- scanset, **7.19.6.2**, **7.24.2.2**
- SCHAR_MAX** macro, **5.2.4.2.1**
- SCHAR_MIN** macro, **5.2.4.2.1**
- SCNcFASTN** macros, **7.8.1**
- SCNcLEASTN** macros, **7.8.1**
- SCNcMAX** macros, **7.8.1**
- SCNcN** macros, **7.8.1**
- SCNcPTR** macros, **7.8.1**
- scope of identifier, **6.2.1**, 6.9.2
- search functions
 - string, **7.21.5**
 - utility, **7.20.5**
 - wide string, **7.24.4.5**
- SEEK_CUR** macro, **7.19.1**, 7.19.9.2
- SEEK_END** macro, **7.19.1**, 7.19.9.2
- SEEK_SET** macro, **7.19.1**, 7.19.9.2
- selection statements, **6.8.4**
- self-referential structure, **6.7.2.3**
- semicolon punctuator (**;**), 6.7, 6.7.2.1, 6.8.3, 6.8.5, 6.8.6
- separate compilation, **5.1.1.1**
- separate translation, **5.1.1.1**
- sequence points, **5.1.2.3**, 6.5, 6.8, 7.1.4, 7.19.6, 7.20.5, 7.24.2, C
- sequencing of statements, **6.8**
- setbuf** function, 7.19.3, 7.19.5.1, **7.19.5.5**
- setjmp** macro, 7.1.3, **7.13.1.1**, 7.13.2.1
- setjmp.h** header, **7.13**
- setlocale** function, **7.11.1.1**, 7.11.2.1
- setvbuf** function, 7.19.1, 7.19.3, 7.19.5.1, 7.19.5.5, **7.19.5.6**
- shall, **4**
- shift expressions, **6.5.7**
- shift sequence, **7.1.1**
- shift states, **5.2.1.2**
- short identifier, character, 5.2.4.1, **6.4.3**
- short int** type, **6.2.5**, 6.3.1.1, 6.7.2, 7.19.6.1, 7.19.6.2, 7.24.2.1, 7.24.2.2
- short int** type conversion, **6.3.1.1**, **6.3.1.3**, **6.3.1.4**, **6.3.1.8**
- SHRT_MAX** macro, **5.2.4.2.1**
- SHRT_MIN** macro, **5.2.4.2.1**
- side effects, **5.1.2.3**, 6.5
- SIG_ATOMIC_MAX** macro, **7.18.3**
- SIG_ATOMIC_MIN** macro, **7.18.3**
- sig_atomic_t** type, **7.14**, 7.14.1.1, 7.18.3
- SIG_DFL** macro, **7.14**, 7.14.1.1
- SIG_ERR** macro, **7.14**, 7.14.1.1
- SIG_IGN** macro, **7.14**, 7.14.1.1
- SIGABRT** macro, **7.14**, 7.20.4.1
- SIGFPE** macro, **7.14**, 7.14.1.1, J.5.17
- SIGILL** macro, **7.14**, 7.14.1.1
- SIGINT** macro, **7.14**
- sign and magnitude, **6.2.6.2**
- sign bit, **6.2.6.2**
- signal** function, **7.14.1.1**, 7.20.4.4
- signal handler, 5.1.2.3, 5.2.3, **7.14.1.1**, 7.14.2.1
- signal handling functions, **7.14.1**
- signal.h** header, **7.14**, 7.26.6

- signaling NaN, **5.2.4.2.2**
- signals, 5.1.2.3, 5.2.3, **7.14.1**
- signbit** macro, **7.12.3.6**, F.3
- signed char** type, **6.2.5**, 7.19.6.1, 7.19.6.2, 7.24.2.1, 7.24.2.2
- signed character, 6.3.1.1
- signed integer types, **6.2.5**, 6.3.1.3, 6.4.4.1
- signed** type conversion, **6.3.1.1**, **6.3.1.3**, **6.3.1.4**, **6.3.1.8**
- signed** types, **6.2.5**, 6.7.2
- significant part, **6.4.4.2**
- SIGSEGV** macro, **7.14**, 7.14.1.1
- SIGTERM** macro, **7.14**
- simple assignment operator (=), **6.5.16.1**
- sin** functions, **7.12.4.6**, **F.9.1.6**
- sin** type-generic macro, **7.22**, **G.7**
- single-byte character, **3.7.1**, 5.2.1.2
- single-byte/wide character conversion functions, **7.24.6.1**
- single-precision arithmetic, **5.1.2.3**
- single-quote escape sequence (\ '), **6.4.4.4**, **6.4.5**
- sinh** functions, **7.12.5.5**, **F.9.2.5**
- sinh** type-generic macro, **7.22**, **G.7**
- SIZE_MAX** macro, **7.18.3**
- size_t** type, 6.5.3.4, **7.17**, 7.18.3, **7.19.1**, 7.19.6.1, 7.19.6.2, **7.20**, **7.21.1**, **7.23.1**, **7.24.1**, 7.24.2.1, 7.24.2.2
- sizeof** operator, 6.3.2.1, 6.5.3, **6.5.3.4**
- snprintf** function, **7.19.6.5**, 7.19.6.12
- sorting utility functions, **7.20.5**
- source character set, 5.1.1.2, **5.2.1**
- source file, **5.1.1.1**
 - name, **6.10.4**, 6.10.8
- source file inclusion, **6.10.2**
- source lines, **5.1.1.2**
- source text, **5.1.1.2**
- space character (' '), 5.1.1.2, **5.2.1**, 6.4, 7.4.1.3, 7.4.1.10, 7.25.2.1.3
- sprintf** function, **7.19.6.6**, 7.19.6.13
- sqrt** functions, **7.12.7.5**, F.3, **F.9.4.5**
- sqrt** type-generic macro, **7.22**
- srand** function, **7.20.2.2**
- sscanf** function, **7.19.6.7**, 7.19.6.14
- standard error stream, 7.19.1, **7.19.3**, 7.19.10.4
- standard headers, 4, **7.1.2**
 - <assert.h>**, **7.2**, B.1
 - <complex.h>**, 5.2.4.2.2, **7.3**, 7.22, 7.26.1, **G.6**, J.5.17
 - <ctype.h>**, **7.4**, 7.26.2
 - <errno.h>**, **7.5**, 7.26.3
 - <fenv.h>**, 5.1.2.3, 5.2.4.2.2, **7.6**, 7.12, F, H
 - <float.h>**, 4, **5.2.4.2.2**, **7.7**, 7.20.1.3, 7.24.4.1.1
 - <inttypes.h>**, **7.8**, 7.26.4
 - <iso646.h>**, 4, **7.9**
 - <limits.h>**, 4, **5.2.4.2.1**, 6.2.5, **7.10**
 - <locale.h>**, **7.11**, 7.26.5
 - <math.h>**, 5.2.4.2.2, 6.5, **7.12**, 7.22, F, **F.9**, J.5.17
 - <setjmp.h>**, **7.13**
 - <signal.h>**, **7.14**, 7.26.6
 - <stdarg.h>**, 4, 6.7.5.3, **7.15**
 - <stdbool.h>**, 4, **7.16**, 7.26.7, H
 - <stddef.h>**, 4, 6.3.2.1, 6.3.2.3, 6.4.4.4, 6.4.5, 6.5.3.4, 6.5.6, **7.17**
 - <stdint.h>**, 4, 5.2.4.2, 6.10.1, 7.8, **7.18**, 7.26.8
 - <stdio.h>**, **7.19**, 7.26.9, F
 - <stdlib.h>**, **7.20**, 7.26.10, F
 - <string.h>**, **7.21**, 7.26.11
 - <tgmath.h>**, **7.22**, **G.7**
 - <time.h>**, **7.23**
 - <wchar.h>**, 7.19.1, **7.24**, 7.26.12, F
 - <wctype.h>**, **7.25**, 7.26.13
- standard input stream, 7.19.1, **7.19.3**
- standard integer types, **6.2.5**
- standard output stream, 7.19.1, **7.19.3**
- standard signed integer types, **6.2.5**
- state-dependent encoding, **5.2.1.2**, 7.20.7
- statements, **6.8**
 - break**, **6.8.6.3**
 - compound, **6.8.2**
 - continue**, **6.8.6.2**
 - do**, **6.8.5.2**
 - else**, **6.8.4.1**
 - expression, **6.8.3**
 - for**, **6.8.5.3**
 - goto**, **6.8.6.1**
 - if**, **6.8.4.1**
 - iteration, **6.8.5**
 - jump, **6.8.6**
 - labeled, **6.8.1**
 - null, **6.8.3**
 - return**, **6.8.6.4**
 - selection, **6.8.4**
 - sequencing, **6.8**
 - switch**, **6.8.4.2**
 - while**, **6.8.5.1**
- static storage duration, **6.2.4**
- static** storage-class specifier, 6.2.2, 6.2.4, **6.7.1**
- static**, in array declarators, 6.7.5.2, **6.7.5.3**
- stdarg.h** header, 4, 6.7.5.3, **7.15**

- stdbool.h** header, 4, **7.16**, 7.26.7, H
- STDC**, **6.10.6**, 6.11.8
- stddef.h** header, 4, 6.3.2.1, 6.3.2.3, 6.4.4.4, 6.4.5, 6.5.3.4, 6.5.6, **7.17**
- stderr** macro, **7.19.1**, 7.19.2, 7.19.3
- stdin** macro, **7.19.1**, 7.19.2, 7.19.3, 7.19.6.4, 7.19.7.6, 7.19.7.7, 7.24.2.12, 7.24.3.7
- stdint.h** header, 4, 5.2.4.2, 6.10.1, 7.8, **7.18**, 7.26.8
- stdio.h** header, **7.19**, 7.26.9, F
- stdlib.h** header, **7.20**, 7.26.10, F
- stdout** macro, **7.19.1**, 7.19.2, 7.19.3, 7.19.6.3, 7.19.7.9, 7.19.7.10, 7.24.2.11, 7.24.3.9
- storage duration, **6.2.4**
- storage order of array, **6.5.2.1**
- storage-class specifiers, **6.7.1**, 6.11.5
- strcat** function, **7.21.3.1**
- strchr** function, **7.21.5.2**
- strcmp** function, 7.21.4, **7.21.4.2**
- strcoll** function, 7.11.1.1, **7.21.4.3**, 7.21.4.5
- strcpy** function, **7.21.2.3**
- strcspn** function, **7.21.5.3**
- streams, **7.19.2**, 7.20.4.3
 - fully buffered, **7.19.3**
 - line buffered, **7.19.3**
 - orientation, **7.19.2**
 - standard error, 7.19.1, **7.19.3**
 - standard input, 7.19.1, **7.19.3**
 - standard output, 7.19.1, **7.19.3**
 - unbuffered, **7.19.3**
- strerror** function, 7.19.10.4, **7.21.6.2**
- strftime** function, 7.11.1.1, 7.23.3, **7.23.3.5**, 7.24.5.1
- strictly conforming program, 4
- string, **7.1.1**
 - comparison functions, **7.21.4**
 - concatenation functions, **7.21.3**
 - conversion functions, 7.11.1.1
 - copying functions, **7.21.2**
 - library function conventions, **7.21.1**
 - literal, 5.1.1.2, 5.2.1, 6.3.2.1, **6.4.5**, 6.5.1, 6.7.8
 - miscellaneous functions, **7.21.6**
 - numeric conversion functions, **7.8.2.3**, **7.20.1**
 - search functions, **7.21.5**
- string handling header, **7.21**
- string.h** header, **7.21**, 7.26.11
- stringizing, **6.10.3.2**, 6.10.9
- strlen** function, **7.21.6.3**
- strncat** function, **7.21.3.2**
- strncmp** function, 7.21.4, **7.21.4.4**
- strncpy** function, **7.21.2.4**
- strpbrk** function, **7.21.5.4**
- strrchr** function, **7.21.5.5**
- strspn** function, **7.21.5.6**
- strstr** function, **7.21.5.7**
- strtod** function, 7.12.11.2, 7.19.6.2, **7.20.1.3**, 7.24.2.2, F.3
- strtof** function, 7.12.11.2, **7.20.1.3**, F.3
- strtoimax** function, **7.8.2.3**
- strtok** function, **7.21.5.8**
- strtol** function, 7.8.2.3, 7.19.6.2, 7.20.1.2, **7.20.1.4**, 7.24.2.2
- strtold** function, 7.12.11.2, **7.20.1.3**, F.3
- strtoll** function, 7.8.2.3, 7.20.1.2, **7.20.1.4**
- strtol** function, 7.8.2.3, 7.19.6.2, 7.20.1.2, **7.20.1.4**, 7.24.2.2
- strtoull** function, 7.8.2.3, 7.20.1.2, **7.20.1.4**
- strtoumax** function, **7.8.2.3**
- struct hack, *see* flexible array member
- structure
 - arrow operator (\rightarrow), **6.5.2.3**
 - content, **6.7.2.3**
 - dot operator (\cdot), **6.5.2.3**
 - initialization, **6.7.8**
 - member alignment, **6.7.2.1**
 - member name space, **6.2.3**
 - member operator (\cdot), 6.3.2.1, **6.5.2.3**
 - pointer operator (\rightarrow), **6.5.2.3**
 - specifier, **6.7.2.1**
 - tag, 6.2.3, **6.7.2.3**
 - type, **6.2.5**, 6.7.2.1
- strxfrm** function, 7.11.1.1, **7.21.4.5**
- subscripting, **6.5.2.1**
- subtraction assignment operator ($\ -=$), **6.5.16.2**
- subtraction operator ($-$), **6.5.6**, F.3, G.5.2
- suffix
 - floating constant, **6.4.4.2**
 - integer constant, **6.4.4.1**
- switch body, **6.8.4.2**
- switch **case** label, 6.8.1, **6.8.4.2**
- switch **default** label, 6.8.1, **6.8.4.2**
- switch** statement, 6.8.1, **6.8.4.2**
- swprintf** function, **7.24.2.3**, 7.24.2.7
- swscanf** function, **7.24.2.4**, 7.24.2.8
- symbols, 3
- syntactic categories, **6.1**
- syntax notation, **6.1**
- syntax rule precedence, **5.1.1.2**
- syntax summary, language, A
- system** function, **7.20.4.6**
- tab characters, **5.2.1**, 6.4

- tag name space, **6.2.3**
- tags, **6.7.2.3**
- tan** functions, **7.12.4.7, F.9.1.7**
- tan** type-generic macro, **7.22, G.7**
- tanh** functions, **7.12.5.6, F.9.2.6**
- tanh** type-generic macro, **7.22, G.7**
- tentative definition, **6.9.2**
- terms, **3**
- text streams, **7.19.2, 7.19.7.11, 7.19.9.2, 7.19.9.4**
- tgamma** functions, **7.12.8.4, F.9.5.4**
- tgamma** type-generic macro, **7.22**
- tgmath.h** header, **7.22, G.7**
- time
 - broken down, **7.23.1, 7.23.2.3, 7.23.3, 7.23.3.1, 7.23.3.3, 7.23.3.4, 7.23.3.5**
 - calendar, **7.23.1, 7.23.2.2, 7.23.2.3, 7.23.2.4, 7.23.3.2, 7.23.3.3, 7.23.3.4**
 - components, **7.23.1**
 - conversion functions, **7.23.3**
 - wide character, **7.24.5**
 - local, **7.23.1**
 - manipulation functions, **7.23.2**
- time** function, **7.23.2.4**
- time.h** header, **7.23**
- time_t** type, **7.23.1**
- tm** structure type, **7.23.1, 7.24.1**
- TMP_MAX** macro, **7.19.1, 7.19.4.3, 7.19.4.4**
- tmpfile** function, **7.19.4.3, 7.20.4.3**
- tmpnam** function, **7.19.1, 7.19.4.3, 7.19.4.4**
- token, **5.1.1.2, 6.4, see also** preprocessing tokens
- token concatenation, **6.10.3.3**
- token pasting, **6.10.3.3**
- tolower** function, **7.4.2.1**
- toupper** function, **7.4.2.2**
- towctrans** function, **7.25.3.2.1, 7.25.3.2.2**
- towlower** function, **7.25.3.1.1, 7.25.3.2.1**
- towupper** function, **7.25.3.1.2, 7.25.3.2.1**
- translation environment, **5, 5.1.1**
- translation limits, **5.2.4.1**
- translation phases, **5.1.1.2**
- translation unit, **5.1.1.1, 6.9**
- trap representation, **6.2.6.1**
- trigonometric functions
 - complex, **7.3.5, G.6.1**
 - real, **7.12.4, F.9.1**
- trigraph sequences, **5.1.1.2, 5.2.1.1**
- true** macro, **7.16**
- trunc** functions, **7.12.9.8, F.9.6.8**
- trunc** type-generic macro, **7.22**
- truncation toward zero, **6.5.5**
- two's complement, **6.2.6.2, 7.18.1.1**
- type category, **6.2.5**
- type conversion, **6.3**
- type definitions, **6.7.7**
- type domain, **6.2.5, G.2**
- type names, **6.7.6**
- type qualifiers, **6.7.3**
- type specifiers, **6.7.2**
- type-generic macro, **7.22, G.7**
- typedef** declaration, **6.7.7**
- typedef** storage-class specifier, **6.7.1, 6.7.7**
- types, **6.2.5**
 - character, **6.7.8**
 - compatible, **6.2.7, 6.7.2, 6.7.3, 6.7.5**
 - complex, **6.2.5, G**
 - composite, **6.2.7**
 - const qualified, **6.7.3**
 - conversions, **6.3**
 - imaginary, **G**
 - restrict qualified, **6.7.3**
 - volatile qualified, **6.7.3**
- UCHAR_MAX** macro, **5.2.4.2.1**
- UINT_FASTN_MAX** macros, **7.18.2.3**
- uint_fastN_t** types, **7.18.1.3**
- UINT_LEASTN_MAX** macros, **7.18.2.2**
- uint_leastN_t** types, **7.18.1.2**
- UINT_MAX** macro, **5.2.4.2.1**
- UINTMAX_C** macro, **7.18.4.2**
- UINTMAX_MAX** macro, **7.8.2.3, 7.8.2.4, 7.18.2.5**
- uintmax_t** type, **7.18.1.5, 7.19.6.1, 7.19.6.2, 7.24.2.1, 7.24.2.2**
- UINTN_C** macros, **7.18.4.1**
- UINTN_MAX** macros, **7.18.2.1**
- uintN_t** types, **7.18.1.1**
- UINTPTR_MAX** macro, **7.18.2.4**
- uintptr_t** type, **7.18.1.4**
- ULLONG_MAX** macro, **5.2.4.2.1, 7.20.1.4, 7.24.4.1.2**
- ULONG_MAX** macro, **5.2.4.2.1, 7.20.1.4, 7.24.4.1.2**
- unary arithmetic operators, **6.5.3.3**
- unary expression, **6.5.3**
- unary minus operator (-), **6.5.3.3, F.3**
- unary operators, **6.5.3**
- unary plus operator (+), **6.5.3.3**
- unbuffered stream, **7.19.3**
- undef** preprocessing directive, **6.10.3.5, 7.1.3, 7.1.4**
- undefined behavior, **3.4.3, 4, J.2**
- underscore character, **6.4.2.1**
- underscore, leading, in identifier, **7.1.3**

- ungetc** function, 7.19.1, **7.19.7.11**, 7.19.9.2, 7.19.9.3
- ungetwc** function, 7.19.1, **7.24.3.10**
- union
 - arrow operator (**->**), **6.5.2.3**
 - content, **6.7.2.3**
 - dot operator (**.**), **6.5.2.3**
 - initialization, **6.7.8**
 - member alignment, **6.7.2.1**
 - member name space, **6.2.3**
 - member operator (**.**), 6.3.2.1, **6.5.2.3**
 - pointer operator (**->**), **6.5.2.3**
 - specifier, **6.7.2.1**
 - tag, 6.2.3, **6.7.2.3**
 - type, **6.2.5**, 6.7.2.1
- universal character name, **6.4.3**
- unqualified type, **6.2.5**
- unqualified version of type, **6.2.5**
- unsigned integer suffix, **u** or **U**, **6.4.4.1**
- unsigned integer types, **6.2.5**, 6.3.1.3, 6.4.4.1
- unsigned** type conversion, **6.3.1.1**, **6.3.1.3**, **6.3.1.4**, **6.3.1.8**
- unsigned** types, **6.2.5**, 6.7.2, 7.19.6.1, 7.19.6.2, 7.24.2.1, 7.24.2.2
- unspecified behavior, **3.4.4**, 4, J.1
- unspecified value, **3.17.3**
- uppercase letter, **5.2.1**
- use of library functions, **7.1.4**
- USHRT_MAX** macro, **5.2.4.2.1**
- usual arithmetic conversions, **6.3.1.8**, 6.5.5, 6.5.6, 6.5.8, 6.5.9, 6.5.10, 6.5.11, 6.5.12, 6.5.15
- utilities, general, **7.20**
 - wide string, **7.24.4**
- va_arg** macro, 7.15, 7.15.1, **7.15.1.1**, 7.15.1.2, 7.15.1.4, 7.19.6.8, 7.19.6.9, 7.19.6.10, 7.19.6.11, 7.19.6.12, 7.19.6.13, 7.19.6.14, 7.24.2.5, 7.24.2.6, 7.24.2.7, 7.24.2.8, 7.24.2.9, 7.24.2.10
- va_copy** macro, 7.15, 7.15.1, 7.15.1.1, **7.15.1.2**, 7.15.1.3
- va_end** macro, 7.1.3, 7.15, 7.15.1, **7.15.1.3**, 7.15.1.4, 7.19.6.8, 7.19.6.9, 7.19.6.10, 7.19.6.11, 7.19.6.12, 7.19.6.13, 7.19.6.14, 7.24.2.5, 7.24.2.6, 7.24.2.7, 7.24.2.8, 7.24.2.9, 7.24.2.10
- va_list** type, **7.15**, 7.15.1.3
- va_start** macro, 7.15, 7.15.1, 7.15.1.1, 7.15.1.2, 7.15.1.3, **7.15.1.4**, 7.19.6.8, 7.19.6.9, 7.19.6.10, 7.19.6.11, 7.19.6.12, 7.19.6.13, 7.19.6.14, 7.24.2.5, 7.24.2.6, 7.24.2.7, 7.24.2.8, 7.24.2.9, 7.24.2.10
- value, **3.17**
- value bits, **6.2.6.2**
- variable arguments, **6.10.3**, 7.15
- variable arguments header, **7.15**
- variable length array, 6.7.5, **6.7.5.2**
- variably modified type, **6.7.5**, 6.7.5.2
- vertical-tab character, **5.2.1**, 6.4
- vertical-tab escape sequence (**\v**), **5.2.2**, 6.4.4.4, 7.4.1.10
- vfprintf** function, 7.19.1, **7.19.6.8**
- vfprintf** function, 7.19.1, 7.19.6.8, **7.19.6.9**
- vwprintf** function, 7.19.1, **7.24.2.5**
- vfwscanf** function, 7.19.1, **7.24.2.6**, 7.24.3.10
- visibility of identifier, **6.2.1**
- void expression, **6.3.2.2**
- void** function parameter, **6.7.5.3**
- void** type, **6.2.5**, 6.3.2.2, 6.7.2
- void** type conversion, **6.3.2.2**
- volatile storage, **5.1.2.3**
- volatile** type qualifier, **6.7.3**
- volatile-qualified type, **6.2.5**, 6.7.3
- vprintf** function, 7.19.1, 7.19.6.8, **7.19.6.10**
- vscanf** function, 7.19.1, 7.19.6.8, **7.19.6.11**
- vsprintf** function, 7.19.6.8, **7.19.6.12**
- vsprintf** function, 7.19.6.8, **7.19.6.13**
- vsscanf** function, 7.19.6.8, **7.19.6.14**
- vswprintf** function, **7.24.2.7**
- vswscanf** function, **7.24.2.8**
- vwprintf** function, 7.19.1, **7.24.2.9**
- vscanf** function, 7.19.1, **7.24.2.10**, 7.24.3.10
- warnings, I
- wchar.h** header, 7.19.1, **7.24**, 7.26.12, F
- WCHAR_MAX** macro, **7.18.3**, **7.24.1**
- WCHAR_MIN** macro, **7.18.3**, **7.24.1**
- wchar_t** type, 3.7.3, 6.4.4.4, 6.4.5, 6.7.8, 6.10.8, **7.17**, 7.18.3, 7.19.6.1, 7.19.6.2, **7.20**, **7.24.1**, 7.24.2.1, 7.24.2.2
- wcrtomb** function, 7.19.3, 7.19.6.2, 7.24.2.2, **7.24.6.3.3**, 7.24.6.4.2
- wcscat** function, **7.24.4.3.1**
- wcschr** function, **7.24.4.5.1**
- wcscmp** function, **7.24.4.4.1**, 7.24.4.4.4
- wcscoll** function, **7.24.4.4.2**, 7.24.4.4.4
- wcscpy** function, **7.24.4.2.1**
- wcscspn** function, **7.24.4.5.2**
- wcsftime** function, 7.11.1.1, **7.24.5.1**
- wcslen** function, **7.24.4.6.1**
- wcsncat** function, **7.24.4.3.2**
- wcsncmp** function, **7.24.4.4.3**

- wcsncpy** function, **7.24.4.2.2**
- wcspbrk** function, **7.24.4.5.3**
- wcsrchr** function, **7.24.4.5.4**
- wcsrtombs** function, **7.24.6.4.2**
- wcsspn** function, **7.24.4.5.5**
- wcsstr** function, **7.24.4.5.6**
- wcstod** function, 7.19.6.2, 7.24.2.2
- wcstod** function, **7.24.4.1.1**
- wcstof** function, **7.24.4.1.1**
- wcstoimax** function, **7.8.2.4**
- wcstok** function, **7.24.4.5.7**
- wcstol** function, 7.8.2.4, 7.19.6.2, 7.24.2.2, **7.24.4.1.2**
- wcstold** function, **7.24.4.1.1**
- wcstoll** function, 7.8.2.4, **7.24.4.1.2**
- wcstombs** function, **7.20.8.2**, 7.24.6.4
- wcstoul** function, 7.8.2.4, 7.19.6.2, 7.24.2.2, **7.24.4.1.2**
- wcstoull** function, 7.8.2.4, **7.24.4.1.2**
- wcstoumax** function, **7.8.2.4**
- wcsxfrm** function, **7.24.4.4.4**
- wctob** function, **7.24.6.1.2**, 7.25.2.1
- wctomb** function, **7.20.7.3**, 7.20.8.2, 7.24.6.3
- wctrans** function, 7.25.3.2.1, **7.25.3.2.2**
- wctrans_t** type, **7.25.1**, 7.25.3.2.2
- wctype** function, 7.25.2.2.1, **7.25.2.2.2**
- wctype.h** header, **7.25**, 7.26.13
- wctype_t** type, **7.25.1**, 7.25.2.2.2
- WEOF** macro, **7.24.1**, 7.24.3.1, 7.24.3.3, 7.24.3.6, 7.24.3.7, 7.24.3.8, 7.24.3.9, 7.24.3.10, 7.24.6.1.1, **7.25.1**
- while** statement, **6.8.5.1**
- white space, 5.1.1.2, **6.4**, 6.10, 7.4.1.10, 7.25.2.1.10
- white-space characters, **6.4**
- wide character, **3.7.3**
 - case mapping functions, **7.25.3.1**
 - extensible, **7.25.3.2**
 - classification functions, **7.25.2.1**
 - extensible, **7.25.2.2**
 - constant, **6.4.4.4**
 - formatted input/output functions, **7.24.2**
 - input functions, **7.19.1**
 - input/output functions, **7.19.1**, **7.24.3**
 - output functions, **7.19.1**
 - single-byte conversion functions, **7.24.6.1**
- wide string, **7.1.1**
- wide string comparison functions, **7.24.4.4**
- wide string concatenation functions, **7.24.4.3**
- wide string copying functions, **7.24.4.2**
- wide string literal, *see* string literal
- wide string miscellaneous functions, **7.24.4.6**
- wide string numeric conversion functions, **7.8.2.4**, **7.24.4.1**
- wide string search functions, **7.24.4.5**
- wide-oriented stream, **7.19.2**
- width, **6.2.6.2**
- WINT_MAX** macro, **7.18.3**
- WINT_MIN** macro, **7.18.3**
- wint_t** type, 7.18.3, 7.19.6.1, **7.24.1**, 7.24.2.1, **7.25.1**
- wmemchr** function, **7.24.4.5.8**
- wmemcmp** function, **7.24.4.4.5**
- wmemcpy** function, **7.24.4.2.3**
- wmemmove** function, **7.24.4.2.4**
- wmemset** function, **7.24.4.6.2**
- wprintf** function, 7.19.1, 7.24.2.9, **7.24.2.11**
- wscanf** function, 7.19.1, 7.24.2.10, **7.24.2.12**, 7.24.3.10
- xor** macro, **7.9**
- xor_eq** macro, **7.9**