A.1.7 Punctuators

(6.4.6) *punctuator:* one of

```
[   ]   (   )   {   }   .    ->
++   --   &   *   +   -   ~   !
/   %   <<   >>   <   >   <=   >=   ==   !=   ^   |   &&   ||
?    :    ::    ;   ...
=   *=   /=   %=   +=   -=   <<=   >>=   &=   ^=   |=
,   #   ##
<:    :>   <%   %>   %:    %:%:
```

A.1.8 Header names

(6.4.7) *header-name:*

< *h-char-sequence* **>**
" *q-char-sequence* **"**

(6.4.7) *h-char-sequence:*

h-char
h-char-sequence h-char

(6.4.7) *h-char:*

any member of the source character set except
the new-line character and >

(6.4.7) *q-char-sequence:*

q-char
q-char-sequence q-char

(6.4.7) *q-char:*

any member of the source character set except
the new-line character and **"**

A.1.9 Preprocessing numbers

(6.4.8) *pp-number:*

digit
. *digit*
pp-number identifier-continue
pp-number **'** *digit*
pp-number **'** *nondigit*
pp-number **e** *sign*
pp-number **E** *sign*
pp-number **p** *sign*
pp-number **P** *sign*
pp-number **.**

A.2 Phrase structure grammar

A.2.1 Expressions

(6.5.1) *primary-expression:*

identifier
constant
string-literal
(*expression* **)**
generic-selection

(6.5.1.1) *generic-selection:*

_Generic (*assignment-expression* **,** *generic-assoc-list* **)**

(6.5.1.1) *generic-assoc-list:*

generic-association
generic-assoc-list **,** *generic-association*

(6.5.1.1) *generic-association:*
type-name **:** *assignment-expression*
default : *assignment-expression*

(6.5.2) *postfix-expression:*
primary-expression
postfix-expression **[** *expression* **]**
postfix-expression **(** *argument-expression-list*$_{\text{opt}}$ **)**
postfix-expression **.** *identifier*
postfix-expression **->** *identifier*
postfix-expression **++**
postfix-expression **--**
compound-literal

(6.5.2) *argument-expression-list:*
assignment-expression
argument-expression-list **,** *assignment-expression*

(6.5.2.5) *compound-literal:*
(*storage-class-specifiers*$_{\text{opt}}$ *type-name* **)** *braced-initializer*

(6.5.2.5) *storage-class-specifiers:*
storage-class-specifier
storage-class-specifiers storage-class-specifier

(6.5.3) *unary-expression:*
postfix-expression
++ *unary-expression*
-- *unary-expression*
unary-operator cast-expression
sizeof *unary-expression*
sizeof (*type-name* **)**
alignof (*type-name* **)**

(6.5.3) *unary-operator:* one of
& * + - ~ !

(6.5.4) *cast-expression:*
unary-expression
(*type-name* **)** *cast-expression*

(6.5.5) *multiplicative-expression:*
cast-expression
multiplicative-expression ***** *cast-expression*
multiplicative-expression **/** *cast-expression*
multiplicative-expression **%** *cast-expression*

(6.5.6) *additive-expression:*
multiplicative-expression
additive-expression **+** *multiplicative-expression*
additive-expression **-** *multiplicative-expression*

(6.5.7) *shift-expression:*
additive-expression
shift-expression **<<** *additive-expression*
shift-expression **>>** *additive-expression*

(6.5.8) *relational-expression:*
shift-expression
relational-expression **<** *shift-expression*
relational-expression **>** *shift-expression*
relational-expression **<=** *shift-expression*
relational-expression **>=** *shift-expression*

(6.5.9) *equality-expression:*
relational-expression
equality-expression **==** *relational-expression*
equality-expression **!=** *relational-expression*

(6.5.10) *AND-expression:*
equality-expression
AND-expression **&** *equality-expression*

(6.5.11) *exclusive-OR-expression:*
AND-expression
exclusive-OR-expression **^** *AND-expression*

(6.5.12) *inclusive-OR-expression:*
exclusive-OR-expression
inclusive-OR-expression **|** *exclusive-OR-expression*

(6.5.13) *logical-AND-expression:*
inclusive-OR-expression
logical-AND-expression **&&** *inclusive-OR-expression*

(6.5.14) *logical-OR-expression:*
logical-AND-expression
logical-OR-expression **||** *logical-AND-expression*

(6.5.15) *conditional-expression:*
logical-OR-expression
logical-OR-expression **?** *expression* **:** *conditional-expression*

(6.5.16) *assignment-expression:*
conditional-expression
unary-expression assignment-operator assignment-expression

(6.5.16) *assignment-operator:* one of
= *= /= %= += -= <<= >>= &= ^= |=

(6.5.17) *expression:*
assignment-expression
expression **,** *assignment-expression*

(6.6) *constant-expression:*
conditional-expression

A.2.2 Declarations

(6.7) *declaration:*
declaration-specifiers init-declarator-list$_{\text{opt}}$ **;**
attribute-specifier-sequence declaration-specifiers init-declarator-list **;**
static_assert-declaration
attribute-declaration

(6.7) *declaration-specifiers:*
declaration-specifier attribute-specifier-sequence$_{\text{opt}}$
declaration-specifier declaration-specifiers

(6.7) *declaration-specifier:*
storage-class-specifier
type-specifier-qualifier
function-specifier

(6.7) *init-declarator-list:*
init-declarator
init-declarator-list **,** *init-declarator*