```
<expresión> -> <expresión de asignación> |
               <expresión>, <expresión de asignación>
<expresión de asignación> -> <expresión condicional> |
                              <expresión unaria> <operador asignación> <expresión de asignación>
<expresión condicional> ->
                              <expresión O lógico> |
                               <expresión O lógico> ? <expresión> : <expresión condicional>
<operador asignación> -> uno de = *= /= %= += -= <<= >>= &= ^= |=
<expresión O lógico> -> <expresión Y lógico> |
                       <expresión O lógico> | | <expresión Y lógico>
<expresión Y lógico> -> <expresión O inclusivo> |
                       <expresión Y lógico> && <expresión O inclusivo>
                              <expresión O excluyente> |
<expresión O inclusivo> ->
                               <expresión O inclusivo> |
                              <expresión O excluyente>
<expresión O excluyente> ->
                              <expresión Y> |
                              <expresión O excluyente> ^ <expresión Y>
<expresión Y> -> <expresión de igualdad> |
                <expresión Y> & <expresión de igualdad>
<expresión de igualdad> -> <expresión relacional> |
                           <expresión de igualdad> == <expresión relacional> |
                           <expresión de igualdad> != <expresión relacional>
<expresión relacional> -> <expresión de corrimiento> |
                         <expresión relacional> < <expresión de corrimiento> |
                          <expresión relacional> > <expresión de corrimiento> |
                          <expresión relacional> <= <expresión de corrimiento> |
                          <expresión relacional> >= <expresión de corrimiento>
<expresión de corrimiento> -> <expresión aditiva> |
                              <expresión de corrimiento> << <expresión aditiva> |
                               <expresión de corrimiento> >> <expresión aditiva>
<expresión aditiva> -> <expresión multiplicativa> |
                       <expresión aditiva> + <expresión multiplicativa> |
                       <expresión aditiva> - <expresión multiplicativa>
```

```
<expresión multiplicativa> -> <expresión de conversión> |
                              <expresión multiplicativa> * <expresión de conversión> |
                              <expresión multiplicativa> / <expresión de conversión> |
                              <expresión multiplicativa> % <expresión de conversión>
<expresión de conversión> -> <expresión unaria> |
                              (<nombre de tipo>) <expresión de conversión>
<expresión unaria> -> <expresión sufijo> |
                      ++ <expresión unaria> |
                      -- <expresión unaria> |
                      <operador unario> <expresión de conversión> |
                      sizeof <expresión unaria> |
                      sizeof (<nombre de tipo>)
<nombre de tipo> está descripto más adelante, en la sección Declaraciones.
<operador unario> -> uno de & * + - ~!
<expresión sufijo> -> <expresión primaria> |
                      <expresión sufijo> [<expresión>] |
                       /* arreglo */ <expresión sufijo> (ta de argumentos>?) |
                       /* invocación */ <expresión sufijo> . <identificador> |
                       <expresión sufijo> -> <identificador> |
                       <expresión sufijo> ++ |
                      <expresión sufijo> --
de argumentos> ->
                              <expresión de asignación> |
                              lista de argumentos>, <expresión de asignación>
<expresión primaria> -> <identificador> |
                      <constante> |
                      <constante cadena> |
                      (<expresión>)
```