AUGUS

Gramáticas

Herbert Rafael Reyes Portillo

201612114

Indice

Contenido

PLY

Expresiones Regulares

- ✓ Decimal = r'\d+\.\d+'
- ✓ Entero= r'\d+'
- ✓ Char = r'\'.\''
- ✓ String= r'\".*?\"'
- ✓ Iden = r'[a-zA-Z_][a-zA-Z_0-9]*'
- ✓ Comentario Simple = r'[#].*\n'

Precedencia Utilizada

Nivel	Operadores	Descripción	Asoci.
1	`() [] -> .`	Acceso a un elemento de un vector y paréntesis	Izquierdas

2	`+ - ! ~`
	`* &`
	`++ --`
	`(cast) sizeof`
		Signo (unario), negación lógica, negación bit a bit
		Acceso a un elemento (unarios): puntero y dirección
		Incremento y decremento (pre y post)
		Conversión de tipo (*casting*) y tamaño de un elemento
			Derechas

3	`* / %`	Producto, división, módulo (resto)	Izquierdas
4	`+ -`	Suma y resta	Izquierdas
5	`>> <<`	Desplazamientos	Izquierdas
6	`< <= >= >`	Comparaciones de superioridad e inferioridad	Izquierdas
7	`== !=`	Comparaciones de igualdad	Izquierdas
8	`&`	Y (*And*) bit a bit (binario)	Izquierdas
9	`^`	O-exclusivo (*Exclusive-Or*) (binario)	Izquierdas
10	`|`	O (*Or*) bit a bit (binario)	Izquierdas
11	`&&`	Y (*And*) lógico	Izquierdas
12	`||`	O (*Or*) lógico	Izquierdas
13	`?:`	Condicional	**Derechas**
14	`= *= /= %= += -= >>= <<= &= ^= |=`	Asignaciones	**Derechas**
15	`,`	Coma	Izquierdas

Símbolos Terminales

Cantidad: 58

1. Break
2. Case
3. Char
4. Continue
5. Default
6. Do
7. Double
8. Else
9. Float
10. For
11. Goto
12. If
13. Int
14. Return
15. Sizeof
16. Struct
17. switch
18. Void
19. While
20. {
21. }
22. (
23.)
24. [
25.]
26. .
27. +
28. –
29. !
30. ~
31. ++
32. –
33. *
34. /
35. %
36. <<
37. >>
38. ==
39. !=
40. &
41. ^
42. |
43. &&
44. ||
45. ?
46. :
47. =
48. *=
49. /=
50. +=
51. -=
52. >>=
53. <<=
54. &=
55. |=
56. ^=
57. ,
58. ;

Símbolos No Terminales Para Gramática Ascendente

No Terminal	Descripcion
S	Inicio Gramatica
Sentencias_G	Lista de las Sentencias Globales
Declaracion	Produce todo tipo de declaraciones
Sentencia_G	Produce todo tipo de sentencia Global
L_Dec	Lista de declaraciones
LASIGNACION	Lista de asignaciones
STRUCT	Produce un struct
Funcion	Produce una función
ASIGNACION	Produce una asignación
OP	Produce todos los tipos de operadores validos para una asignacion
ATRIBUTOS	Produce una lista de atributos
ATRIBUTO	Produce todos los atributos necesarios del struct
Items Struct	Produce los diversos tipos que puede llevar adentro un struct
IntesStruct2	Produce los ítems que puede llevar la instancia de un struct
Tipos	Produce los tipos que el lenguaje reconoce
ACCESO_Struct	Produccion de los tipos de accesos que se puede hacer en un struct
Parametros	Produccion de los parámetros que puede poseer una función
SS_F	Lista se sentencias de Funcion
S_F	Produccion que contiene los tipos de instrucciones que puede realizarse dentro de una función
INSTANCIA	Instrucción de instancia de un struct
IF	Instrucción de sentencia de if
LLAMADA	Realiza llamada a una función
BREAK	Produccion de break
CONTINUE	Produccion de continue
SWTICH	Produccion del swtich
CASOS	Lista de producciones de CASO
CASO	Instrucción de un caso de switch
DEFAULT	Instrucción default del switch
DO_WHILE	Instrucción do while
WHILE	Instrucción while
Return	Instrucción return
GOTO	Instrucción de goto
Etiqueta	Instrucción de etiqueta
EXP	Produccion que genera todo tipo de expresión valida en el lenguaje
ELEMENTS	Lista de expresiones

Grámatica Ascendente

Gramatica Ascendente
S → Sentencias_G
S → Empty
Sentencias_G --> Sentencias_G Sentencia_G
Sentencias_G --> Sentencia_G
Sentencia_G --> error
Sentencia_G --> Declaracion | LASIGNACION pyc | STRUCT pyc | Funcion
Declaracion --> Tipos L_Dec pyc
L_Dec --> L_Dec coma Dec
L_Dec --> Dec
Dec --> iden
Dec --> iden asigna EXP
Dec --> iden cor1 cor2 asigna EXP
Dec --> iden LACCESO
Dec --> iden LACCESO asigna EXP
LACCESO --> LACCESO cor1 EXP cor2
LACCESO --> cor1 EXP cor2
LASIGNACION --> LASIGNACION coma ASIGNACION
LASIGNACION --> ASIGNACION
ASIGNACION --> iden
ASIGNACION --> iden OP EXP
ASIGNACION --> iden LACCESO
ASIGNACION --> iden LACCESO OP EXP
ASIGNACION --> ACCESO_STRUCT OP EXP
ASIGNACION --> ACCESO_STRUCT
OP --> andbasigna | divasigna | masasigna | menosasigna | modasigna | orbasigna | porasigna | shiftizqasigna | shiftderasigna | xorbasinga | asigna
STRUCT --> t_struct iden llav1 ATRIBUTOS llav2
ATRIBUTOS --> ATRIBUTOS ATRIBUTO

ATRIBUTOS --> ATRIBUTO
ATRIBUTO --> Tipos LItemsStruct pyc
LItemsStruct --> LItemsStruct coma ItemsStruct
LItemsStruct --> ItemsStruct
ItemsStruct --> iden
ItemsStruct --> iden LACCESO
ItemsStruct2 --> iden
ItemsStruct2 --> iden LACCESO
ACCESO_STRUCT --> ItemsStruct2 punto ItemsStruct2
Tipos --> t_char
Tipos --> t_int
Tipos --> t_double
Tipos --> t_float
Funcion --> Tipos iden par1 Parametros par2 BLOQUE
Funcion --> t_void iden par1 Parametros par2 BLOQUE
Parametros -->
Parametros --> Parametros coma Parametro
Parametros --> Parametro
SS_F --> SS_F S_F
SS_F --> S_F
S_F --> error
S_F --> Declaracion **| LASIGNACION pyc** **| INSTANCIA pyc** **| STRUCT pyc** **| IF** **| LLAMADA pyc** **| BREAK pyc** **| CONTINUE pyc** **| SWITCH** **| DO_WHILE pyc** **| WHILE** **| RETURN pyc** **| ETIQUETA** **| GOTO pyc** **| FOR**
INSTANCIA --> t_struct iden iden LACCESO
INSTANCIA --> t_struct iden iden
BLOQUE --> llav1 SS_F llav2
BLOQUE --> llav1 llav2
FOR --> t_for par1 INICIO EXP pyc INCDC BLOQUE
INICIO --> Tipos Dec pyc
INICIO --> ASIGNACION pyc
INICIO --> pyc
INCDC --> EXP par2
INCDC --> par2

ETIQUETA --> iden bipunto
GOTO --> t_goto iden
DO_WHILE --> t_do BLOQUE t_while par1 EXP par2
WHILE --> t_while par1 EXP par2 BLOQUE
RETURN --> t_return EXP
RETURN --> t_return
SWITCH --> t_switch par1 EXP par2 llav1 CASOS llav2
SWITCH --> t_switch par1 EXP par2 llav1 CASOS DEFAULT llav2
CASOS --> CASOS CASO
CASOS --> CASO
CASO --> t_case EXP bipunto SS_F
CASO --> t_case EXP bipunto
DEFAULT --> t_default bipunto SS_F
DEFAULT --> t_default bipunto
BREAK --> t_break
CONTINUE --> t_continue
IF --> t_if par1 EXP par2 BLOQUE
IF --> t_if par1 EXP par2 BLOQUE t_else IF
IF --> t_if par1 EXP par2 BLOQUE t_else BLOQUE
Parametro --> Tipos iden
Parametro --> Tipos iden LACCESO
LLAMADA --> iden par1 ELEMENTS par2
LLAMADA --> iden par1 par2
EXP --> EXP mas EXP | EXP menos EXP | EXP por EXP | EXP division EXP | EXP modulo EXP
EXP --> EXP mayor EXP | EXP mayori EXP | EXP menor EXP | EXP menori EXP | EXP igual EXP | EXP diferente EXP
EXP --> EXP and EXP | EXP or EXP
EXP --> iden LACCESO
EXP --> string | entero | decimal | char
EXP --> iden
EXP --> par1 EXP par2

EXP --> EXP shiftizq EXP **| EXP shiftder EXP** **| EXP andb EXP** **| EXP xorb EXP** **| EXP orb EXP**
EXP --> mas EXP **| menos EXP** **| not EXP** **| notb EXP** **| andb EXP**
EXP --> incremento EXP **| decremento EXP**
EXP --> EXP condicional EXP bipunto EXP
EXP --> EXP incremento **| EXP decrement**
EXP --> llav1 ELEMENTS llav2
ELEMENTS --> ELEMENTS coma EXP
ELEMENTS --> EXP
EXP --> LLAMADA
EXP --> ACCESO_STRUCT
EXP --> t_sizeof par1 EXP par2