

Gramática del Lenguaje de Programación ALFA

1	<programa>	::=	main { <declaraciones> <funciones> <sentencias> }
2	<declaraciones>	::=	<declaracion>
3			<declaracion> <declaraciones>
4	<declaracion>	::=	<clase> <identificadores> ;
5	<clase>	::=	<clase_escalor>
6			<clase_puntero>
7			<clase_vector>
8			<clase_conjunto>
9	<clase_escalor>	::=	<tipo>
10	<tipo>	::=	int
11			boolean
12			float
13	<clase_puntero>	::=	<tipo> *
14			<clase_puntero> *
15	<clase_vector>	::=	array <tipo> [<constante_entera>]
16			array <tipo> [<constante_entera> ,
17	<clase_conjunto>	::=	set of <constante_entera>
18	<identificadores>	::=	<identificador>
19			<identificador> , <identificadores>
20	<funciones>	::=	<funcion> <funciones>
21			
22	<funcion>	::=	function <tipo> <identificador>
23			(<parametros_funcion>) { <declaraciones_funcion>
24	<parametros_funcion>	::=	<parametro_funcion> <resto_parametros_funcion>
25	<resto_parametros_funcion>	::=	; <parametro_funcion> <resto_parametros_funcion>
26			
27	<parametro_funcion>	::=	<tipo> <identificador>
28	<declaraciones_funcion>	::=	<declaraciones>
29			
30	<sentencias>	::=	<sentencia>
31			<sentencia> <sentencias>
32	<sentencia>	::=	<sentencia_simple> ;
33			<bloque>
34	<sentencia_simple>	::=	<asignacion>
35			<lectura>
36			<escritura>
37			<liberacion>
38			<retorno_funcion>
39			<operacion_conjunto>
40	<bloque>	::=	<condicional>
41			<bucle>
42			<seleccion>
43	<asignacion>	::=	<identificador> = <exp>
44			<elemento_vector> = <exp>
45			<acceso> = <exp>
46			<identificador> = malloc
47			<identificador> = & <identificador>
48	<elemento_vector>	::=	<identificador> [<exp>]
49			<identificador> [<exp> , <exp>]
50	<condicional>	::=	if (<exp>) { <sentencias> }
51			if (<exp>) { <sentencias> } else { <sentencias> }
52	<bucle>	::=	while (<exp>) { <sentencias> }
53			for (<identificador> = <exp> ; <exp>) {
54	<lectura>	::=	scanf <identificador>

55			scanf <elemento_vector>
56	<escritura>	::=	printf <exp>
57			cprintf <identificador>
58	<liberacion>	::=	free <identificador>
59	<acceso>	::=	* <identificador>
60			* <acceso>
61	<retorno_funcion>	::=	return <exp>
62	<seleccion>	::=	switch (<exp>) { <casos_seleccion> }
63	<casos_seleccion>	::=	<casos_estandar> <caso_defecto>
64	<casos_estandar>	::=	<caso_estandar>
65			<casos_estandar> <caso_estandar>
66	<caso_estandar>	::=	case <constante_entera> : <sentencias>
67	<caso_defecto>	::=	default <sentencias>
68	<operacion_conjunto>	::=	union (<identificador> ,<identificador> ,
69			<identificador>)
70			intersection (<identificador> , <identificador> ,
71			<identificador>)
72	<exp>	::=	add (<exp> , <identificador>)
73			clear (<identificador>)
74			<exp> + <exp>
75			<exp> - <exp>
76			<exp> / <exp>
77			<exp> * <exp>
78			- <exp>
79			<exp> && <exp>
80			<exp> <exp>
81			! <exp>
82			<identificador>
83			<constante>
84			(<exp>)
85			(<comparacion>)
86			<acceso>
87			<elemento_vector>
88			size (<identificador>)
89	<lista_expresiones>	::=	contains (<exp> , <identificador>)
90			<identificador> (<lista_expresiones>)
91	<resto_lista_expresiones>	::=	<exp> <resto_lista_expresiones>
92			, <exp> <resto_lista_expresiones>
93	<comparacion>	::=	<exp> == <exp>
94			<exp> != <exp>
95			<exp> <= <exp>
96			<exp> >= <exp>
97			<exp> < <exp>
98			<exp> > <exp>
99	<constante>	::=	<constante_logica>
100			<constante_entera>
101			<constante_real>
102	<constante_logica>	::=	true
103			false
104	<constante_entera>	::=	<numero>
105	<numero>	::=	<digito>
106			<numero> <digito>
107	<constante_real>	::=	<constante_entera> . <constante_entera>
108	<identificador>	::=	<letra>
109			<letra> <cola_identificador>
110	<cola_identificador>	::=	<alfanumerico>
111			<alfanumerico> <cola_identificador>
112	<alfanumerico>	::=	<letra>
113			<digito>
114	<letra>	::=	a b ... z A B ... Z

Consideraciones Adicionales

- El lenguaje permite incluir comentarios entre los caracteres // y el final de la línea (son comentarios de una sola línea).
- Los identificadores se limitan a una longitud de 100 caracteres.

NOTA: Solamente las producciones resaltadas en gris son objeto del compilador de este curso