

Manual Tecnico

Yefer Rodrigo Miguel Alvarado Tzul

Carné: 201731163

Universidad San Carlos de Guatemala

Centro Universitario de Occidente (CUNOC)

División de Ciencias de la Ingeniería

Organizaciones de Lenguajes y Compiladores 2

Ing. Moises Granados

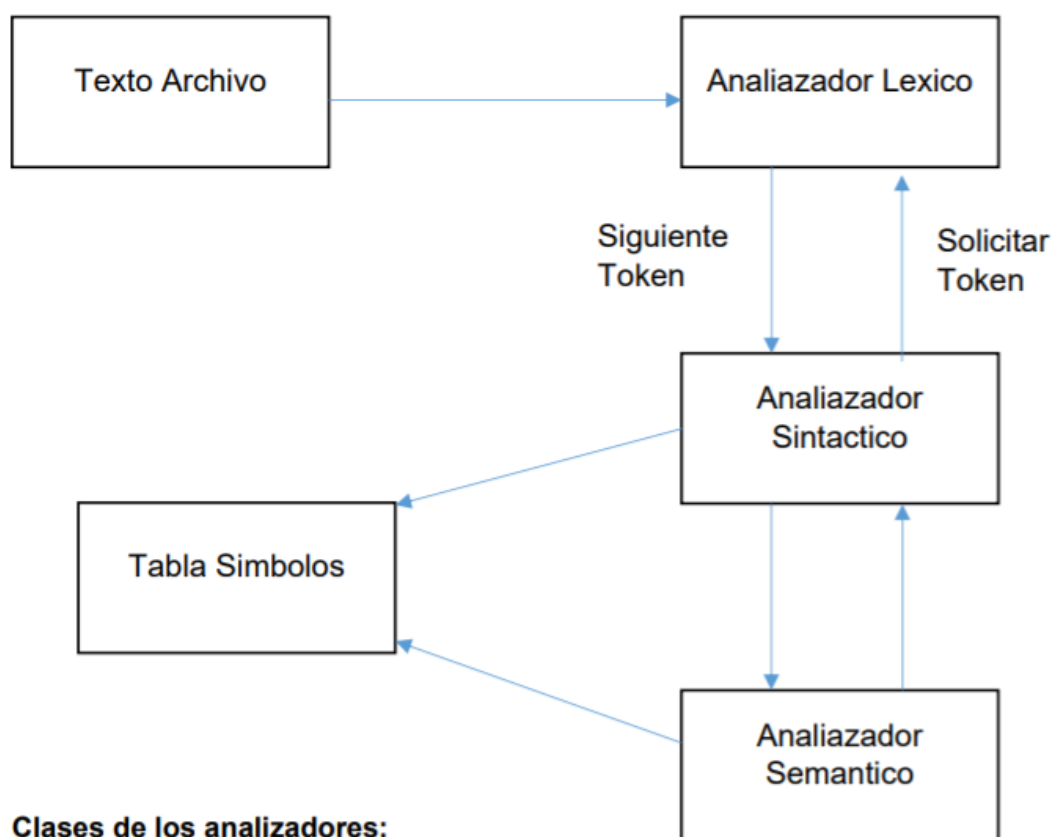
10 de septiembre de 2021

Manual Tecnico

DESCRIPCIÓN DEL PROGRAMA

El programa cuenta con un analizador léxico y sintáctico. Estos se usan para determinar los objetos que se crearán para integración del componente del programa. Cuenta con dos analizadores sintácticos ya que, uno se encarga de los errores y otro de recolectar los datos para el ast. Y un analizador léxico para darle color a las palabras.

Tenemos un archivo de lectura el cual nos encargamos de guardar luego.



Clases de los analizadores:

AnalizadorLexicoCode

AnalizadorSintacticoCode

AnalizadorLexico

AnalizadorSintactico

AnalizadorLexicoTS

AnalizadorSintacticoTS

Implementación del IDE

El ide cuenta con una ventana, en esta se agrega el código fuente, para posteriormente éste sea analizada, y en caso de que existan errores abrirá una ventana de errores.

El programa fue hecho con Netbeans IDE.

Se utilizo el Lenguaje de Java en su Version 11 y Kotlin.

El codigo fuente se encuentra en las carpetas src.

GRAMÁTICA

Analizador Léxico

Terminales

PISTA, IDENTIFICADOR, EXTIENDE, COMA, PUNTO_COMA, ENTERO, DOBLE, BOOLEAN, CHARACTER_RSV, CADENA_RSV, IGUAL_IGUAL, DIFERENTE, MAYOR_Q, MENOR_Q, MAYOR_IGUAL, MENOR_IGUAL, IS_NULL, AND, NAND, OR, NOR, XOR, NOT, MAS, MENOS, POR, DIVISION, MODULO, POTENCIA, KEEP, VAR, IGUAL, NUMERO, DECIMAL, CADENA, TRUE, FALSE, CHARACTER, MAS_IGUAL, MAS_MAS, MENOS_MENOS, ARREGLO, CORCHETE_A, CORCHETE_C, LLAVE_A, LLAVE_C, SI, SINO, PA_A, PA_C, SWITCH, CASO, SALIR, DEFAULT, PARA, MIENTRAS, HACER, CONTINUAR, RETORNAR, VOID, REPRODUCIR, ESPERAR, ORDENAR, ASCENDENTE, DESCENDENTE, PARES, IMPARES, PRIMOS, SUMARIZAR, LONGITUD, MENSAJE, PRINCIPAL, DO, RE, MI, FA, SOL, LA, DO_S, RE_S, FA_S, SOL_S, LA_S, SINO_SI, INDENT, DEDENT.

Analizador Sintactico

no Terminales

INICIO, blanco, codigo, struct_pista, struct_extiende, extiende_state, struc_code, struct_declaracion_var, boolean keep_state, valor_declaracion_operacion, valor, struc_asig_var, tipo_dato, struct_operador_logico, valor_operacion, struct_declaracion_array, dim_arreglo, valor_arreglo, valores_array, struct_si, struct_sino, struct_switch, state_tab, struct_asig_arreglo, struct_para, struct_mientras, struct_hacer, struct_funcion, struct_procedimiento, struct_reproducir, struct_esperar, struct_ordenar, struct_sumarizar, struct_longitud, struct_mensaje, struct_principal, strunt_llamada_funcion, struct_llamada_arreglo, parametros, state_params,

STRUCT_PISTA ::= pista identificador:a EXTIENDE_STATE:b llave_a
 STRUCT_CODE:c llave_c
 |error:e identificador:a EXTIENDE_STATE:b llave_a
 STRUCT_CODE:c llave_c
 |pista error:e EXTIENDE_STATE llave_a STRUCT_CODE
 llave_c
 |pista identificador:a error:e llave_a STRUCT_CODE:c llave_c
 |pista identificador EXTIENDE_STATE llave_a error:e llave_c
 ;

EXTIENDE_STATE ::= extiende STRUCT_EXTIENDE:a
 |
 |error:e STRUCT_EXTIENDE ;

STRUCT_EXTIENDE ::= STRUCT_EXTIENDE:a coma identificador:b
 |identificador:a
 |STRUCT_EXTIENDE:a error:e identificador
 |STRUCT_EXTIENDE:a coma error:e
 |error:e coma identificador
 ;

STRUCT_CODE ::= STRUCT_CODE:a COMENTARIOS
 |STRUCT_CODE:a
 STRUCT_DECLARACION_VARIABLE_GLOBAL:b

```

|STRUCT_CODE:a STRUCT_ASIGNACION_VAR:b
|STRUCT_CODE:a
STRUCT_DECLARACION_ARREGLO_GLOBAL:b
|STRUCT_CODE:a STRUCT_ASIGNACION_ARREGLO:b
|STRUCT_CODE:a STRUCT_FUNCION_PROC:b
|STRUCT_CODE:a STRUCT_METODO_PRINCIPAL:b
|;

STRUCT_CODE_MAIN ::= STRUCT_CODE_MAIN COMENTARIOS
|STRUCT_CODE_MAIN
STRUCT_DECLARACION_VARIABLE_GLOBAL
|STRUCT_CODE_MAIN
STRUCT_ASIGNACION_VAR
|STRUCT_CODE_MAIN
STRUCT_DECLARACION_ARREGLO_GLOBAL
|STRUCT_CODE_MAIN
STRUCT_ASIGNACION_ARREGLO
|STRUCT_CODE_MAIN STRUCT_FUNCION_PROC
|;

```

```

STRUCT_DECLARACION_VARIABLE_GLOBAL ::=
DECLARACION_VARIABLE_GLOBAL_STATE:d ASIGNACION_STATE:a punto_coma

```

|DECLARACION_VARIABLE_GLOBAL_STATE error:e punto_coma

;

DECLARACION_VARIABLE_GLOBAL_STATE ::=

DECLARACION_VARIABLE_GLOBAL_STATE:d coma identificador:a

|KEEP_STATE:a

var:m TIPO_DATO_STATE:b identificador:c

|KEEP_STATE:a

var error:e identificador

|KEEP_STATE:a

var TIPO_DATO_STATE error:e

|DECLARACION_VARIABLE_GLOBAL_STATE coma error:e

;

KEEP_STATE ::= keep

| ;

TIPO_DATO_STATE ::= entero

|doble

|boolean_rsv

|cadena_rsv

|caracter_rsv ;

ASIGNACION_STATE ::= igual VALOR_STATE:a

|

|igual error:e

|error:e VALOR_STATE

;

STRUCT_ASIGNACION_VAR ::= identificador:a igual VALOR_STATE:b

punto_coma

|identificador:a mas_igual

VALOR_STATE:b punto_coma

|identificador:a mas_mas punto_coma

|identificador:a menos_menos

punto_coma

|identificador error:e VALOR_STATE

punto_coma

|identificador igual error:e punto_coma

|identificador error:e punto_coma

;

STRUCT_DECLARACION_VARIABLE ::=

DECLARACION_VARIABLE_STATE:d ASIGNACION_STATE:a punto_coma

|DECLARACION_VARIABLE_STATE error:e punto_coma

	error:e ASIGNACION_STATE:a
punto_coma	
	;
DECLARACION_VARIABLE_STATE ::=	
DECLARACION_VARIABLE_STATE:d coma identificador:a	
	var:m TIPO_DATO_STATE:b
identificador:c	
	error:e TIPO_DATO_STATE
identificador	
	var error:e identificador
	var TIPO_DATO_STATE error:e
	error:e coma identificador
DECLARACION_VARIABLE_STATE coma error:e	
	;

STRUCT_DECLARACION_ARREGLO_GLOBAL ::=

DECLARACION_ARREGLO_GLOBAL_STATE:d STRUCT_DIMENSION_ARREGLO:b

ASIGNACION_ARREGLO_STATE:a punto_coma

|DECLARACION_ARREGLO_GLOBAL_STATE error:e

ASIGNACION_ARREGLO_STATE punto_coma

|DECLARACION_ARREGLO_GLOBAL_STATE STRUCT_DIMENSION_ARREGLO

error:e punto_coma

;

DECLARACION_ARREGLO_GLOBAL_STATE ::=

DECLARACION_ARREGLO_GLOBAL_STATE:d coma identificador:a

|KEEP_STATE:a

var TIPO_DATO_STATE:b arreglo:m identificador:c

|DECLARACION_ARREGLO_GLOBAL_STATE error:e identificador

|DECLARACION_ARREGLO_GLOBAL_STATE coma error:e

|KEEP_STATE

var TIPO_DATO_STATE error:e identificador

;

STRUCT_DIMENSION_ARREGLO ::= STRUCT_DIMENSION_ARREGLO:d
corchete_a VALOR_STATE:a corchete_c

|STRUCT_DIMENSION_ARREGLO:d corchete_a corchete_c
|corchete_a VALOR_STATE:a
corchete_c
|corchete_a corchete_c
|corchete_a error:e corchete_c
|error:e corchete_a corchete_c

|STRUCT_DIMENSION_ARREGLO:e corchete_a error corchete_c
;

ASIGNACION_ARREGLO_STATE ::= igual VALOR_ARREGLO_STATE:a
|
|error:e

VALOR_ARREGLO_STATE
|igual error:e
;

VALOR_ARREGLO_STATE ::= llave_a:m VALOR_ARREGLO_MULTI_STATE:a
llave_c

|llave_a:m VALOR_UNIT_ARREGLO:a

llave_c

|llave_a:m llave_c

|llave_a error:e llave_c

;

VALOR_ARREGLO_MULTI_STATE ::= VALOR_ARREGLO_MULTI_STATE:d

coma llave_a:m VALOR_ARREGLO_MULTI_STATE:a llave_c

|VALOR_ARREGLO_MULTI_STATE:d coma llave_a:m VALOR_UNIT_ARREGLO:a

llave_c

|llave_a:m

VALOR_ARREGLO_MULTI_STATE:a llave_c

|llave_a:m

VALOR_UNIT_ARREGLO:a llave_c

|VALOR_ARREGLO_MULTI_STATE coma llave_a error:e llave_c

|VALOR_ARREGLO_MULTI_STATE error:e llave_a VALOR_ARREGLO_MULTI_STATE

llave_c

|VALOR_ARREGLO_MULTI_STATE error:e llave_a VALOR_UNIT_ARREGLO llave_c

|llave_a error:e llave_c

;

VALOR_UNIT_ARREGLO ::= VALOR_UNIT_ARREGLO:d coma VALOR_STATE:a
 |VALOR_STATE:a
 |VALOR_UNIT_ARREGLO coma error:e
 |error:e coma VALOR_STATE
 ;

STRUCT_ASIGNACION_ARREGLO ::= identificador:a
 DIMESION_POS_ARREGLO:b igual VALOR_STATE:c punto_coma
 |identificador:a error:e igual
 VALOR_STATE:c punto_coma
 |identificador:a
 DIMESION_POS_ARREGLO:b error:e VALOR_STATE:c punto_coma
 |identificador:a
 DIMESION_POS_ARREGLO:b igual error:e punto_coma
 ;

DIMESION_POS_ARREGLO ::= DIMESION_POS_ARREGLO:d corchete_a
 VALOR_STATE:a corchete_c
 |corchete_a VALOR_STATE:a corchete_c
 |DIMESION_POS_ARREGLO corchete_a
 error:e corchete_c
 |corchete_a error:e corchete_c

;

STRUCT_DECLARACION_ARREGLO ::=

DECLARACION_ARREGLO_STATE:d STRUCT_DIMENSION_ARREGLO:b

ASIGNACION_ARREGLO_STATE:a punto_coma

|error:e

STRUCT_DIMENSION_ARREGLO ASIGNACION_ARREGLO_STATE punto_coma

|DECLARACION_ARREGLO_STATE error:e ASIGNACION_ARREGLO_STATE

punto_coma

|DECLARACION_ARREGLO_STATE STRUCT_DIMENSION_ARREGLO error:e

punto_coma

;

DECLARACION_ARREGLO_STATE ::=

DECLARACION_ARREGLO_STATE:d coma identificador:a

|var:m TIPO_DATO_STATE:b

arreglo identificador:c

|DECLARACION_ARREGLO_STATE error:e identificador

|DECLARACION_ARREGLO_STATE coma error:e

|var error:e arreglo identificador

|var TIPO_DATO_STATE error:e

identificador

|var TIPO_DATO_STATE arreglo

error:e

;

CONDICIONAL_SI_SINO_SI ::= CONDICIONAL_SI_SINO_SI_STATE:a

STRUCT_CONDICIONAL_SINO:b

|CONDICIONAL_SI_SINO_SI_STATE:a

|error:e

STRUCT_CONDICIONAL_SINO

;

CONDICIONAL_SI_SINO_SI_STATE ::= CONDICIONAL_SI_SINO_SI_STATE:a

STRUCT_CONDICIONAL_SINO_SI:b

|STRUCT_CONDICIONAL_SI:a

|error:e

STRUCT_CONDICIONAL_SINO_SI

;

STRUCT_CONDICIONAL_SI ::= si:m pa_a VALOR_CODICIONAL:a pa_c llave_a

SENTENCIAS_CODIGO:b llave_c

|si pa_a error:e pa_c llave_a

SENTENCIAS_CODIGO llave_c

|si pa_a VALOR_CODICIONAL pa_c error:e

SENTENCIAS_CODIGO llave_c

;

STRUCT_CONDICIONAL_SINO_SI ::= sino_si:m pa_a VALOR_CODICIONAL:a pa_c

llave_a SENTENCIAS_CODIGO:b llave_c

|sino_si pa_a error:e pa_c llave_a

SENTENCIAS_CODIGO:b llave_c

|sino_si pa_a VALOR_CODICIONAL

pa_c error:e SENTENCIAS_CODIGO:b llave_c

;

STRUCT_CONDICIONAL_SINO ::= sino:m llave_a SENTENCIAS_CODIGO:a llave_c

|sino:m error:e

SENTENCIAS_CODIGO:a llave_c

;

```

STRUCT_CONDICIONAL_SWITCH ::= switch_rsv:m pa_a VALOR_STATE:a
pa_c llave_a CASO_STATE:b llave_c

|switch_rsv pa_a error:e pa_c

llave_a CASO_STATE llave_c

|switch_rsv pa_a

VALOR_STATE pa_c llave_a error:e llave_c

;

```

```

CASO_STATE ::= STRUCT_CASO default_rsv:m dos_puntos
SENTENCIAS_CODIGO:a

|STRUCT_CASO:a

|STRUCT_CASO default_rsv error:e

SENTENCIAS_CODIGO

;

```

```

STRUCT_CASO ::= STRUCT_CASO:c caso:m VALOR_CASO:a dos_puntos
SENTENCIAS_CODIGO_CASO:b

|caso:m VALOR_CASO:a dos_puntos

SENTENCIAS_CODIGO_CASO:b

```

|error:e caso VALOR_CASO dos_puntos

SENTENCIAS_CODIGO_CASO

|STRUCT_CASO caso error:e dos_puntos

SENTENCIAS_CODIGO_CASO

|STRUCT_CASO caso VALOR_CASO error:e

SENTENCIAS_CODIGO_CASO

|caso error:e dos_puntos SENTENCIAS_CODIGO_CASO

|caso VALOR_CASO error:e SENTENCIAS_CODIGO_CASO

;

VALOR_CASO ::= numero:a

|cadena:a

|caracter:a

|decimal:a ;

STRUCT_PARA ::= para:m pa_a ASIG_PARA_STATE:a punto_coma

COND_PARA_STATE:b punto_coma SENTENCIAS_PARA_STATE:c pa_c llave_a

SENTENCIAS_CODIGO:d llave_c

|para pa_a error:e punto_coma COND_PARA_STATE

punto_coma SENTENCIAS_PARA_STATE pa_c llave_a SENTENCIAS_CODIGO llave_c

|para pa_a ASIG_PARA_STATE punto_coma error:e
punto_coma SENTENCIAS_PARA_STATE pa_c llave_a SENTENCIAS_CODIGO llave_c

|para pa_a ASIG_PARA_STATE punto_coma
COND_PARA_STATE punto_coma error:e pa_c llave_a SENTENCIAS_CODIGO llave_c

;

ASIG_PARA_STATE ::= var:m TIPO_DATO_STATE:c identificador:a igual
VALOR_STATE:b

|identificador:a igual VALOR_STATE:b

|var:m error:e identificador:a igual VALOR_STATE

|var:m TIPO_DATO_STATE:c error:e igual

VALOR_STATE

|var:m TIPO_DATO_STATE:c identificador:a error:e

VALOR_STATE

|var:m TIPO_DATO_STATE:c identificador:a igual

error:e

|error:e igual VALOR_STATE

|identificador error:e VALOR_STATE

|identificador:a igual error:e

;

COND_PARA_STATE ::= VALOR_CODICIONAL:a ;


```

STRUCT_HACER_MIENTRAS ::= hacer:m llave_a SENTENCIAS_CODIGO:b llave_c
mientras pa_a VALOR_CODICIONAL:a pa_c
                                |hacer:m llave_a SENTENCIAS_CODIGO:b
llave_c mientras pa_a error:e pa_c
                                |hacer:m llave_a SENTENCIAS_CODIGO:b
llave_c error:e pa_a VALOR_CODICIONAL:a pa_c
                                ;

```

```

STRUCT_FUNCION_PROC ::= identificador:a pa_a STRUCT_PARAMS_STATE:b
pa_c llave_a SENTENCIAS_CODIGO_FUNC_PROC:c llave_c
                                |TIPO_DATO_STATE_OPTIONAL:d
identificador:a pa_a STRUCT_PARAMS_STATE:b pa_c llave_a
SENTENCIAS_CODIGO_FUNC_PROC:c llave_c
                                |keep identificador:a pa_a
STRUCT_PARAMS_STATE:b pa_c llave_a SENTENCIAS_CODIGO_FUNC_PROC:c
llave_c

```

|keep TIPO_DATO_STATE_OPTIONAL:d

identificador:a pa_a STRUCT_PARAMS_STATE:b pa_c llave_a

SENTENCIAS_CODIGO_FUNC_PROC:c llave_c

|identificador:a pa_a error:e pa_c llave_a

SENTENCIAS_CODIGO_FUNC_PROC llave_c

|identificador:a pa_a

STRUCT_PARAMS_STATE:b pa_c error:e llave_c

|TIPO_DATO_STATE_OPTIONAL:d error:e

pa_a STRUCT_PARAMS_STATE:b pa_c llave_a

SENTENCIAS_CODIGO_FUNC_PROC:c llave_c

|TIPO_DATO_STATE_OPTIONAL:d

identificador:a pa_a error:e pa_c llave_a SENTENCIAS_CODIGO_FUNC_PROC:c llave_c

|keep identificador:a pa_a error:e pa_c llave_a

SENTENCIAS_CODIGO_FUNC_PROC:c llave_c

|keep TIPO_DATO_STATE_OPTIONAL:d

error:e pa_a STRUCT_PARAMS_STATE:b pa_c llave_a

SENTENCIAS_CODIGO_FUNC_PROC:c llave_c

|keep TIPO_DATO_STATE_OPTIONAL:d

identificador:a pa_a error:e pa_c llave_a SENTENCIAS_CODIGO_FUNC_PROC:c llave_c

;

TIPO_DATO_STATE_OPTIONAL ::= entero

|doble

|boolean_rsv

|cadena_rsv

|caracter_rsv

|void_rsv ;

STRUCT_PARAMS_STATE ::= STRUCT_PARAMS:a

| ;

STRUCT_PARAMS ::= STRUCT_PARAMS:a coma TIPO_DATO_STATE:b

identificador:c

|TIPO_DATO_STATE:a identificador:b

|error:e coma TIPO_DATO_STATE:b identificador

|STRUCT_PARAMS:a coma error:e identificador

|STRUCT_PARAMS:a coma TIPO_DATO_STATE:b

error:e

|error:e identificador

|TIPO_DATO_STATE error:e

;

STRUCT_REPRODUCIR ::= reproducir:e pa_a NOTA_STATE:a coma

OCTAVA_STATE:b coma TIEMPO_MS_STATE:c coma CANAL_STATE:d pa_c


```

|reproducir pa_a error:e coma OCTAVA_STATE
coma TIEMPO_MS_STATE coma CANAL_STATE pa_c

|reproducir pa_a NOTA_STATE coma error:e
coma TIEMPO_MS_STATE coma CANAL_STATE pa_c

|reproducir pa_a NOTA_STATE coma
OCTAVA_STATE coma error:e coma CANAL_STATE pa_c

|reproducir pa_a NOTA_STATE coma
OCTAVA_STATE coma TIEMPO_MS_STATE coma error:e

;

```

```

NOTA_STATE ::= do_rsv

```

```

|do_s

```

```

|re

```

```

|re_s

```

```

|mi

```

```

|fa

```

```

|fa_s

```

```

|sol

```

```

|sol_s

```

```

|la

```

```

|la_s

```

```

|si ;

```

```

OCTAVA_STATE ::= VALOR_STATE:a ;

```

TIEMPO_MS_STATE ::= VALOR_STATE:a ;

CANAL_STATE ::= VALOR_STATE:a ;

STRUCT_ESPERAR ::= esperar pa_a VALOR_STATE:a coma VALOR_STATE:b pa_c
|esperar pa_a error:e coma VALOR_STATE pa_c
|esperar pa_a VALOR_STATE:a coma error:e pa_c
;

STRUCT_ORDENAR ::= ordenar pa_a VALOR_ORDENAR_STATE:a pa_c
|ordenar pa_a error:e pa_c
;

VALOR_ORDENAR_STATE ::= identificador:a coma TIPO_ORDEN:b
|VALOR_ARREGLO_STATE:a coma
TIPO_ORDEN:b
|error:e coma TIPO_ORDEN
|identificador coma error:e

|VALOR_ARREGLO_STATE error:e

TIPO_ORDEN

|VALOR_ARREGLO_STATE coma error:e

;

TIPO_ORDEN ::= ascendente

|descendente

|pares

|impares

|primos ;

STRUCT_SUMARIZAR ::= sumarizar:b pa_a identificador:a pa_c

|sumarizar:b pa_a VALOR_ARREGLO_STATE:a pa_c

|sumarizar pa_a error:e pa_c

;

STRUCT_LONGITUD ::= longitud:b pa_a identificador:a pa_c

|longitud:b pa_a VALOR_ARREGLO_STATE:a pa_c

|longitud:b pa_a cadena:a pa_c

|longitud pa_a error:e pa_c

;

STRUCT_MENSAJE ::= mensaje:b pa_a VALOR_STATE:a pa_c

|mensaje:b pa_a error:e pa_c

;

STRUCT_METODO_PRINCIPAL ::= principal:b pa_a pa_c llave_a

SENTENCIAS_CODIGO:a llave_c

|principal:b error:e llave_c

|principal:b error:e llave_a

SENTENCIAS_CODIGO:a llave_c

;

```

VALOR_DATA ::=      numero:a

                    |decimal:a

                    |true_rsv:a

                    |false_rsv:a

                    |caracter:a

                    |cadena:a

                    |identificador:a

                    |caracter_special_comilla_simple:a

                    |caracter_special_hashtag:a

                    |caracter_special_r:a

                    |caracter_special_t:a

                    |caracter_special_n:a

                    |STRUCT_LLAMADA_FUNCION:a

                    |STRUCT_LLAMADA_ARREGLO:a

                    |STRUCT_REPRODUCIR:a

                    |STRUCT_ORDENAR:a

                    |STRUCT_SUMARIZAR:a

                    |STRUCT_LONGITUD:a

                    ;

```

```

STRUCT_LLAMADA_FUNCION ::=  identificador:a pa_a

PARAMAS_STATE_LLAMADA:b pa_c

                        |identificador:a pa_a pa_c

```

|identificador:a pa_a error:e pa_c ;

PARAMAS_STATE_LLAMADA ::= PARAMAS_STATE_LLAMADA:a coma
VALOR_STATE:b

|VALOR_STATE:a

|error:e coma VALOR_STATE

|PARAMAS_STATE_LLAMADA:a

coma error:e

;

STRUCT_LLAMADA_ARREGLO ::= identificador:a DIMESION_POS_ARREGLO:b

|identificador:a error:e

|error:e DIMESION_POS_ARREGLO

;

OPERACION_ARITMETICA ::= OPERACION_ARITMETICA:a mas:c

OPERACION_ARITMETICA:b

|OPERACION_ARITMETICA:a menos:c

OPERACION_ARITMETICA:b

|OPERACION_ARITMETICA:a por:c

OPERACION_ARITMETICA:b

|OPERACION_ARITMETICA:a div:c

OPERACION_ARITMETICA:b

	OPERACION_ARITMETICA:a modulo:c
OPERACION_ARITMETICA:b	
	OPERACION_ARITMETICA:a potencia:c
OPERACION_ARITMETICA:b	
	pa_a OPERACION_ARITMETICA:a pa_c
	VALOR_DATA:a
	pa_a error:e pa_c
	error:e OPERACION_ARITMETICA
	;

OPERACION_RACIONAL ::=	OPERACION_RACIONAL:a igual_igual:c
OPERACION_RACIONAL:b	
	OPERACION_RACIONAL:a diferente:c
OPERACION_RACIONAL:b	
	OPERACION_RACIONAL:a mayor_q:c
OPERACION_RACIONAL:b	
	OPERACION_RACIONAL:a menor_q:c
OPERACION_RACIONAL:b	
	OPERACION_RACIONAL:a mayor_igual:c
OPERACION_RACIONAL:b	
	OPERACION_RACIONAL:a menor_igual:c
OPERACION_RACIONAL:b	

|is_null OPERACION_RACIONAL:a

|OPERACION_ARITMETICA:a

|is_null error:e

;

OPERACION_LOGICA ::= OPERACION_LOGICA:a and:c OPERACION_LOGICA:b

|OPERACION_LOGICA:a nand:c

OPERACION_LOGICA:b

|OPERACION_LOGICA:a or:c

OPERACION_LOGICA:b

|OPERACION_LOGICA:a nor:c

OPERACION_LOGICA:b

|OPERACION_LOGICA:a xor:c

OPERACION_LOGICA:b

|not OPERACION_LOGICA:a

|not error:e

|OPERACION_RACIONAL:a ;

VALOR_STATE ::= OPERACION_LOGICA:a

;

VALOR_CODICIONAL ::= OPERACION_LOGICA:a ;

SENTENCIAS_CODIGO ::= SENTENCIAS_CODIGO:d COMENTARIOS

|SENTENCIAS_CODIGO:d

STRUCT_DECLARACION_VARIABLE:b

|SENTENCIAS_CODIGO:d

STRUCT_ASIGNACION_VAR:b

|SENTENCIAS_CODIGO:d

STRUCT_DECLARACION_ARREGLO:b

|SENTENCIAS_CODIGO:d

STRUCT_ASIGNACION_ARREGLO:b

|SENTENCIAS_CODIGO:d

CONDICIONAL_SI_SINO_SI:b

|SENTENCIAS_CODIGO:d

STRUCT_CONDICIONAL_SWITCH:b

|SENTENCIAS_CODIGO:d STRUCT_PARA:b

|SENTENCIAS_CODIGO:d

STRUCT_MIENTRAS:b

|SENTENCIAS_CODIGO:d

STRUCT_HACER_MIENTRAS:b

	SENTENCIAS_CODIGO:d
STRUCT_REPRODUCIR:b punto_coma	
	SENTENCIAS_CODIGO:d
STRUCT_ESPERAR:b punto_coma	
	SENTENCIAS_CODIGO:d
STRUCT_ORDENAR:b punto_coma	
	SENTENCIAS_CODIGO:d
STRUCT_SUMARIZAR:b punto_coma	
	SENTENCIAS_CODIGO:d
STRUCT_LONGITUD:b punto_coma	
	SENTENCIAS_CODIGO:d
STRUCT_MENSAJE:b punto_coma	
	SENTENCIAS_CODIGO:d
STRUCT_LLAMADA_FUNCION:b punto_coma	
	SENTENCIAS_CODIGO:d
STRUCT_RETORNAR:b punto_coma	
	SENTENCIAS_CODIGO:d salir:a punto_coma
	SENTENCIAS_CODIGO:d continuar:a
punto_coma	
	;
STRUCT_RETORNAR ::= retornar:m VALOR_CODICIONAL:a	
	retornar error:e
	;

SENTENCIAS_CODIGO_CASO ::= SENTENCIAS_CODIGO:a ;

SENTENCIAS_CODIGO_FUNC_PROC ::= SENTENCIAS_CODIGO:a ;

Lenguaje de Comunicación

No Terminales

INICIO, CODE, STATE_SOLICITUD, STATE_CONT_SOLICITUD,
STATE_OPT_SOLICITUD, STATE_NUEVA_SOLICITUD, STATE_DATOS,
NOTA_STATE, STATE_OTROS, STATE_LISTA_LISTAS,
STATE_CONT_LISTA_LISTAS, STATE_LISTA_PISTAS, STATE_CONT_LISTA_PISTAS,
STATE_LISTA_PISTAS_ONLY, STATE_ALEATORIO,
STATE_CONTE_LISTA_PISTAS_ONLY, STATE_PISTA, STATE_CONT_PISTA,
STRUCT_PISTA, STATE_NOTES_CHANNEL, STATE_EMPTY_LISTA_LISTAS,
STATE_EMPTY_LISTA_PISTAS, STATE_EMPTY_LISTA_PISTAS_ONLY,
STATE_EMPTY_PISTA, STATE_EMPTY_NOTES_CHANNEL;

Terminales

solicitud_a, solicitud_c, lista, pista, tipo_a, tipo_c, nombre_a, nombre_c, listas_a, listas_c,
lista_a, lista_c, nombre, pistas, pista_ab, duracion, aleatorio, si_rsv, no, pistas_a, pistas_c,
canal_ab, numero_rsv, canal_c, nota_ab, frecuencia, pista_c, datos_a, datos_c, canal_a,
nota_a, nota_c, octava_a, octava_c, duracion_a, duracion_c, cerrar, igual, cadena,
identificador, numero, decimal, do_rsv, re, mi, fa, sol, la, do_s, re_s, fa_s, sol_s, la_s, si;

Producciones

INICIO ::= CODE:a ;

CODE ::= STATE_SOLICITUD:a

|STATE_OTROS:a ;

STATE_SOLICITUD ::= solicitud_a STATE_CONT_SOLICITUD:a solicitud_c

|error:e solicitud_c

;

STATE_CONT_SOLICITUD ::= tipo_a lista tipo_c STATE_OPT_SOLICITUD:b

|tipo_a pista tipo_c STATE_OPT_SOLICITUD:b

|tipo_a identificador:a tipo_c STATE_NUEVA_SOLICITUD:b

|error:e

;

STATE_OPT_SOLICITUD ::= nombre_a cadena:a nombre_c

|;

STATE_NUEVA_SOLICITUD ::= STATE_NUEVA_SOLICITUD:r STATE_DATOS:a

|STATE_DATOS:a

|error:e STATE_DATOS

;

STATE_DATOS ::= datos_a

canal_a numero:a canal_c

nota_a NOTA_STATE:b nota_c

octava_a numero:c octava_c

duracion_a numero:d duracion_c

datos_c

|error:e datos_c

;

NOTA_STATE ::= do_rsv

|do_s

|re

|re_s

|mi

|fa

|fa_s

|sol

|sol_s

|la

|la_s

|si ;

STATE_OTROS ::= STATE_LISTA_LISTAS:a STATE_LISTA_PISTAS:b

|STATE_LISTA_PISTAS_ONLY:a STATE_PISTA:b

|error:e STATE_LISTA_PISTAS

|error:e STATE_PISTA

|STATE_LISTA_LISTAS error:e

|STATE_LISTA_PISTAS_ONLY error:e

;

STATE_LISTA_LISTAS ::= listas_a

STATE_EMPTY_LISTA_LISTAS:a

listas_c

|error:e listas_c

;

STATE_CONT_LISTA_LISTAS ::= STATE_CONT_LISTA_LISTAS:r lista_a nombre
igual cadena:a pistas igual numero:b cerrar

|lista_a nombre igual cadena:a pistas igual numero:b cerrar

|error:e cerrar

;

STATE_EMPTY_LISTA_LISTAS ::= STATE_CONT_LISTA_LISTAS:a

|;

STATE_LISTA_PISTAS ::= STATE_LISTA_PISTAS:r lista_a nombre igual cadena:a
aleatorio igual STATE_ALEATORIO:b cerrar

STATE_EMPTY_LISTA_PISTAS:c

lista_c

|lista_a nombre igual cadena:a aleatorio igual STATE_ALEATORIO:b

cerrar

STATE_EMPTY_LISTA_PISTAS:c

lista_c

|error:e cerrar

STATE_EMPTY_LISTA_PISTAS:c

lista_c

|error:e lista_c

;

STATE_ALEATORIO ::= si_rsv

|no ;

STATE_EMPTY_LISTA_PISTAS ::= STATE_CONT_LISTA_PISTAS:a

|;

STATE_CONT_LISTA_PISTAS ::= STATE_CONT_LISTA_PISTAS:r pista_ab nombre
igual cadena:a duracion igual numero:b cerrar

|pista_ab nombre igual cadena:a duracion igual numero:b cerrar

|error:e cerrar

;

STATE_LISTA_PISTAS_ONLY ::= pistas_a

STATE_EMPTY_LISTA_PISTAS_ONLY:a

pistas_c

|error:e pistas_c

;

STATE_CONTE_LISTA_PISTAS_ONLY ::= STATE_CONTE_LISTA_PISTAS_ONLY:r

pista_ab nombre igual cadena:a duracion igual numero:b cerrar

|pista_ab nombre igual cadena:a duracion igual numero:b cerrar

|error:e cerrar

;

STATE_EMPTY_LISTA_PISTAS_ONLY ::= STATE_CONTE_LISTA_PISTAS_ONLY:a

|;

STATE_PISTA ::= STATE_PISTA:r pista_ab nombre igual cadena:a cerrar

STATE_EMPTY_PISTA:b

pista_c

|pista_ab nombre igual cadena:a cerrar

STATE_EMPTY_PISTA:b

pista_c

|error:e cerrar

STATE_EMPTY_PISTA:b

pista_c

|error:e pista_c

;

STATE_CONT_PISTA ::= STATE_CONT_PISTA:r STRUCT_PISTA:a

|STRUCT_PISTA:a

|error:e STRUCT_PISTA

;

STATE_EMPTY_PISTA ::= STATE_CONT_PISTA:a

|;

STRUCT_PISTA ::= canal_ab numero_rsv igual numero:a cerrar

STATE_EMPTY_NOTES_CHANNEL:b

canal_c

|error:e cerrar

STATE_EMPTY_NOTES_CHANNEL:b

canal_c

|error:e canal_c

;

STATE_NOTES_CHANNEL ::= STATE_NOTES_CHANNEL:r nota_ab duracion igual
numero:a frecuencia igual decimal:b cerrar

|nota_ab duracion igual numero:a frecuencia igual decimal:b cerrar

|error:e cerrar

;

STATE_EMPTY_NOTES_CHANNEL ::= STATE_NOTES_CHANNEL:a

;

**** Process exited - Return Code: 0 ****

Press Enter to exit terminal