

## MANUAL TÉCNICO

### Detalles Herramientas Usadas en el Software:

- Angular-cli version 11
- Node js version 14
- Jison (Modificado)
- Angular Material
- d3-graphviz (npm i d3-graphviz@2.6.1) -como api graficadora-

### Detalles del Jison

/\* description: Parses and evaluates mathematical expressions. \*/

### Analizador Lexico

Token	Lexema
SYNTAX	Syntax
INIT	Initial_Sim
NoTERMINALINIT	No_Terminal
Comentario 1 linea	"#".*
Comentario bloque	["/"]["***"][^*]*["*"]+([/^*][^*]*["***"]+)*["/"]
ALLLETTERS	"[aA-zZ]"
ALLNUMBERS	"[0-9]"
{	{
}	}
[	[
]	]
(	(
)	)
<	<
=	=
-	-

;	;
:	:
+	+
*	*
?	?
RESERVED_WORD	['][^\\']+[']
WISON	Wison
LEX	Lex
¿	¿
TERMINALINIT	Terminal
TERMINAL	"\$_"([^\s;<-])+
NOTERMINAL	"%_"([^\s;<-])+

## ANALIZADOR SINTÁCTICO

Empieza con s.

s -> WISON ¿ p

p -> LEX { : t y

y -> SYNTAX { { : sy

t -> er : }

er -> var\_re

var\_re -> var var\_re

var -> TERMINALINIT TERMINAL < - expresion\_re

;

expresion\_re -> expresion

expresion -> ( expresion ) fer expresion  
| [ expresion ] fer expresion  
| (TERMINAL) expresion  
| ALLNUMBERS fer expresion  
| ALLLETTERS fer expresion  
| RESERVED\_WORD expresion  
| ALFANUM expresion  
| e

fer -> \*  
| +  
| ?  
| e

sy -> not\_re ini : } } ? WISON EOF

not\_re -> not not\_re  
| e  
;

not -> NOTERMINALINIT NOTERMINAL ;

ini -> INIT NOTERMINAL ; vars\_re

vars\_re -> vars vars\_re  
| e

vars -> NOTERMINAL < = ef

ef -> proc ef\_re

ef\_re -> '|' proc ef\_re  
| e

proc -> proc\_re

proc\_re -> n proc\_re  
| e

```
n : TERMINAL
|NOTERMINAL
```

## GENERADOR PARSERS

Codigo:

```
%lex
%%
"\s+ /*skip*/";
```

Producciones Extra

```
".* /*LEXIC ERROR*/";
"<<EOF>> return 'EOF'";
%start Token_Iniciador
%%
```

Reglas de produccion