Memoria de la Práctica de Procesadores de Lenguajes: Analizador Sintáctico

Diego José Abengózar Vilar, Alejandro García Castellanos, Ignacio Javier Encinas Ramos

${\rm Grupo}~82$

November 26, 2019

Índice

1	Dise	eño del Analizador Sintáctico	
	1.1	Gramática	
	1.2	Autómata Reconocedor de Prefijos Viables	
		1.2.1 Estados del autómata	
	1.3	Conflictos y Tabla de Decisión	
	1.4	Errores	
2	Ane	exo de Pruebas	

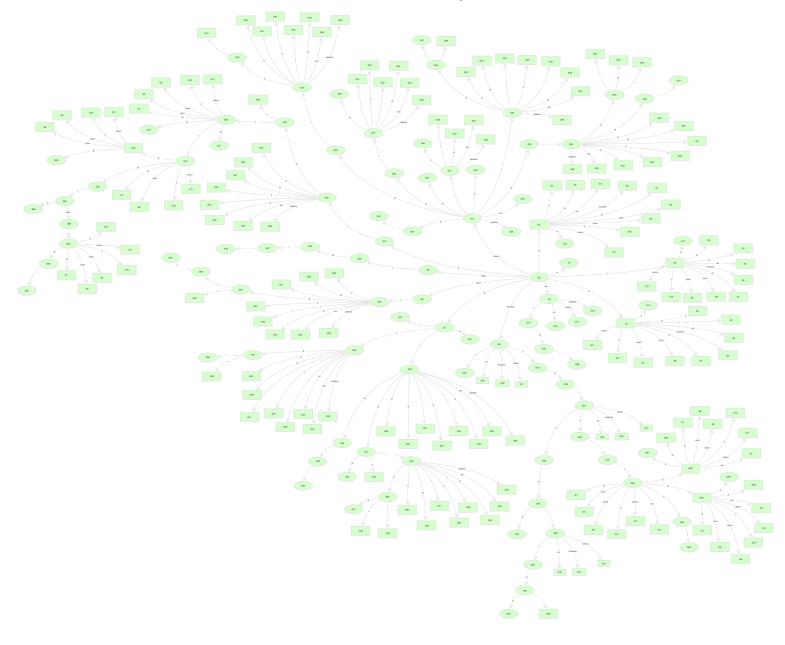
1 Diseño del Analizador Sintáctico

1.1 Gramática

```
Terminales = \{ ; \{ \} id ent cadena ( ) + < ! = | = var int \}
boolean string print input, return function if else }
NoTerminales = \{ PDTFT1AKCSLMQS1GXEURVS2 \}
Axioma = P
Producciones = \{
     P \rightarrow D P
      P \rightarrow F P
      P \rightarrow S P
      D \rightarrow var T id ;
      T \rightarrow int
      T \rightarrow string
      T \rightarrow boolean
      F \rightarrow function T1 id (A) \{C\}
      T1 \rightarrow \lambda
      T1 \rightarrow T
     A \rightarrow T id K
      A \rightarrow \lambda
     K \rightarrow \lambda
     K \rightarrow T id K
      C \rightarrow D C
      C \rightarrow S C
      C \rightarrow \lambda
      S \rightarrow id L E;
      S \rightarrow id (M);
      S \rightarrow print (E);
      S \rightarrow input \ (id);
      S \rightarrow if (E) S1
      S \rightarrow return X;
      L \rightarrow |=
      L \rightarrow =
     M \rightarrow E Q
     M \rightarrow \lambda
      Q \rightarrow \lambda
      Q \rightarrow E Q
      S1 \rightarrow \{S2\}G
      S1 \rightarrow S
      G \rightarrow else \{ S2 \}
      G \rightarrow \lambda
     X \rightarrow E
     X \rightarrow \lambda
     E \rightarrow E < U
      E \rightarrow U
      U \rightarrow U + R
      U \rightarrow R
```

```
egin{array}{ll} R & 
ightarrow & ! & V \ R & 
ightarrow & V \ V & 
ightarrow & (E) \ V & 
ightarrow & id \ V & 
ightarrow & id \ (M) \ V & 
ightarrow & ent \ V & 
ightarrow & cadena \ S2 & 
ightarrow & S2 \ S2 & 
ightarrow & S2 \ P & 
ightarrow & \lambda \ \end{array}
```

1.2 Autómata Reconocedor de Prefijos Viables



1.2.1 Estados del autómata

```
S_0=\{P1 \rightarrow \bullet P, P \rightarrow \bullet DP, P \rightarrow \bullet SP, P \rightarrow \bullet, D \rightarrow \bullet var T id;,
         F \rightarrow \bullet \text{ function } T1 \text{ id}(A)\{C\}, S \rightarrow \bullet \text{ id } L E;, S \rightarrow \bullet \text{ id}(M);
         S \rightarrow \bullet \text{ print}(E); , S \rightarrow \bullet \text{ input}(id); , S \rightarrow \bullet \text{ if}(E) S1,
         S \rightarrow \bullet return X;
S_1 = \{P1 \rightarrow P \bullet \}
S_2 = \{P \rightarrow D \bullet P, P \rightarrow \bullet DP, P \rightarrow \bullet SP, P \rightarrow \bullet , D \rightarrow \bullet var T id;,
         F \rightarrow \bullet function T1 id(A){C}, S \rightarrow \bullet id L E;, S \rightarrow \bullet id(M);
         S \rightarrow \bullet print(E); S \rightarrow \bullet input(id); S \rightarrow \bullet if(E) S1,
         S \rightarrow \bullet return X;
S_3=\{P \rightarrow F \bullet P, P \rightarrow \bullet DP, P \rightarrow \bullet FP, P \rightarrow \bullet SP, P \rightarrow \bullet,
         D \rightarrow \bullet \text{ var } T \text{ id}; F \rightarrow \bullet \text{ function } T1 \text{ id}(A) \{C\},
         S \rightarrow \bullet \text{ id } L E;, S \rightarrow \bullet \text{ id}(M);, S \rightarrow \bullet \text{ print}(E);
         S \rightarrow \bullet \text{ input (id)}; S \rightarrow \bullet \text{ if (E)} S1, S \rightarrow \bullet \text{ return } X;
S_4=\{P \rightarrow S \bullet P, P \rightarrow \bullet DP, P \rightarrow \bullet FP, P \rightarrow \bullet SP, P \rightarrow \bullet,
         D \rightarrow \bullet \text{ var } T \text{ id};, F \rightarrow \bullet \text{ function } T1 \text{ id}(A)\{C\},
         S \rightarrow \bullet id L E;, S \rightarrow \bullet id (M);, S \rightarrow \bullet print (E);,
         S \ \rightarrow \ \bullet \ input(id); \, , \ S \ \rightarrow \ \bullet \ if(E) \ S1 \, , \ S \ \rightarrow \ \bullet \ return \ X; \}
S_5 = \{D \rightarrow var \bullet T id;, T \rightarrow \bullet int, T \rightarrow \bullet string, T \rightarrow \bullet boolean\}
S_6=\{F \rightarrow function \bullet T1 \ id(A)\{C\}, \ T1 \rightarrow \bullet, \ T1 \rightarrow \bullet T,
        T \rightarrow \bullet \text{ int}, T \rightarrow \bullet \text{ string}, T \rightarrow \bullet \text{ boolean}
S_7 = \{S \rightarrow id \bullet L E; , S \rightarrow id \bullet (M); , L \rightarrow |=, L \rightarrow \bullet =\}
S_8 = \{S \rightarrow print \bullet (E); \}
S_9 = \{S \rightarrow input \bullet (id); \}
S_{10} = \{S \rightarrow if \bullet (E) S1\}
S_{11}\!\!=\!\!\{S \ \rightarrow\! \texttt{return} \ \bullet X; \,, \ X \ \rightarrow \ \bullet, \ X \ \rightarrow \ \bullet \ E, \ E \ \rightarrow \ \bullet \ E < U, \ E \ \rightarrow \ \bullet \ U,
         U \rightarrow \bullet U + R, U \rightarrow \bullet R, R \rightarrow \bullet ! V, R \rightarrow \bullet V, V \rightarrow \bullet (E),
         V \rightarrow \bullet \text{ id}, V \rightarrow \bullet \text{ id}(M), V \rightarrow \bullet \text{ ent}, V \rightarrow \bullet \text{ cadena}
S_{12} = \{P \rightarrow DP \bullet\}
S_{13} = \{P \rightarrow FP \bullet \}
S_{14} = \{P \rightarrow SP \bullet \}
S_{15}=\{D \rightarrow var \ T \bullet id;\}
S_{16} = \{T \rightarrow int \bullet \}
S_{17} = \{T \rightarrow string \bullet \}
S_{18} = \{T \rightarrow boolean \bullet \}
S_{19} = \{F \rightarrow function \ T1 \bullet id(A)\{C\}\}\
S_{20} {=} \{T1 \ \rightarrow T \ \bullet\}
S_{21} = \{L \rightarrow = \bullet\}
S_{22} = \{S \rightarrow id \ (\bullet M); , M \rightarrow \bullet E Q, M \rightarrow \bullet, E \rightarrow \bullet E < U, E \rightarrow \bullet U,
         U \rightarrow \bullet U + R, U \rightarrow \bullet R, R \rightarrow \bullet ! V, R \rightarrow \bullet V, V \rightarrow \bullet (E),
         V \rightarrow \bullet \text{ id}, V \rightarrow \bullet \text{ id}(M), V \rightarrow \bullet \text{ ent}, V \rightarrow \bullet \text{ cadena}
S_{23} = \{L \rightarrow |= \bullet\}
S_{24} = \{S \rightarrow print \ (\bullet E); , E \rightarrow \bullet E < U, E \rightarrow \bullet U, U \rightarrow \bullet U + R,
         U \rightarrow \bullet R, R \rightarrow \bullet ! V, R \rightarrow \bullet V, V \rightarrow \bullet (E), V \rightarrow \bullet id,
         V \rightarrow \bullet id(M), V \rightarrow \bullet ent, V \rightarrow \bullet cadena
S_{25} = \{S \rightarrow input \ ( \bullet id ) \}
S_{26} = \{S \rightarrow if \ (\bullet E) \ S1, \ E \rightarrow \bullet E < U, \ E \rightarrow \bullet U, \ U \rightarrow \bullet U + R,
         U \ \rightarrow \ \bullet \ R, \ R \ \rightarrow \ \bullet \ ! \ V, \ R \ \rightarrow \ \bullet \ V, \ V \ \rightarrow \ \bullet \ (E) \,, \ V \ \rightarrow \ \bullet \ id \,,
         V \rightarrow \bullet id(M), V \rightarrow \bullet ent, V \rightarrow \bullet cadena)
S_{27} = \{S \rightarrow return \ X \bullet ; \}
S_{28} = \{X \rightarrow E \bullet, E \rightarrow E \bullet < U\}
S_{29} = \{E \rightarrow U \bullet, U \rightarrow U \bullet + R\}
S_{30} = \{U \rightarrow R \bullet \}
S_{31} = \{R \rightarrow ! \bullet V, V \rightarrow \bullet (E), V \rightarrow \bullet id, V \rightarrow \bullet id(M),
        V \rightarrow \bullet \text{ ent}, V \rightarrow \bullet \text{ cadena}
S_{32} = \{R \rightarrow V \bullet \}
S_{33}=\{V \rightarrow (\bullet E), E \rightarrow \bullet E < U, E \rightarrow \bullet U, U \rightarrow \bullet U + R,
```

```
U \rightarrow \bullet R, R \rightarrow \bullet ! V, R \rightarrow \bullet V, V \rightarrow \bullet (E), V \rightarrow \bullet id,
         V \rightarrow \bullet id(M), V \rightarrow \bullet ent, V \rightarrow \bullet cadena
S_{34} = \{V \rightarrow id \bullet, V \rightarrow id \bullet (M)\}
S_{35} = \{V \rightarrow ent \bullet \}
S_{36} = \{V \rightarrow cadena \bullet \}
S_{37}=\{D \rightarrow var \ T \ id \bullet;\}
S_{38} = \{F \rightarrow function T1 id \bullet (A)\{C\}\}\
S_{39}=\{S \rightarrow id L \bullet E, E \rightarrow \bullet E < U, E \rightarrow \bullet U, U \rightarrow \bullet U + R,
         U \rightarrow \bullet R, R \rightarrow \bullet ! V, R \rightarrow \bullet V, V \rightarrow \bullet (E), V \rightarrow \bullet id,
         V \rightarrow \bullet id(M), V \rightarrow \bullet ent, V \rightarrow \bullet cadena
S_{40} = \{S \rightarrow id (M \bullet); \}
S_{41}=\{M \rightarrow E \bullet Q, E \rightarrow E \bullet < U, Q \rightarrow \bullet, Q \rightarrow \bullet, EQ\}
S_{42} = \{S \rightarrow print(E \bullet);, E \rightarrow E \bullet < U\}
S_{43} = \{S \rightarrow input(id \bullet);\}
S_{44} = \{S \rightarrow if(E \bullet) S1, E \rightarrow E \bullet < U\}
S_{45} = \{S \rightarrow return X; \bullet \}
S_{46}=\{E \rightarrow E < \bullet U, U \rightarrow \bullet R, U \rightarrow \bullet U + R, R \rightarrow \bullet ! V, R \rightarrow \bullet V\}
         V \rightarrow \bullet (E), V \rightarrow \bullet id, V \rightarrow \bullet id (M), V \rightarrow \bullet ent, V \rightarrow \bullet cadena}
S_{47}=\{U \rightarrow U + \bullet R, R \rightarrow \bullet ! V, V \rightarrow \bullet (E), V \rightarrow \bullet id(M),
         V \rightarrow \bullet \text{ ent}, V \rightarrow \bullet \text{ cadena}
S_{48} = \{R \rightarrow ! V \bullet \}
S_{49} = \{V \rightarrow (E \bullet), E \rightarrow E \bullet < U\}
S_{50} = \{ V \rightarrow id \ ( \bullet \ M ) \ , \ M \rightarrow \bullet \ E \ Q, \ M \rightarrow \bullet \ , \ E \rightarrow \bullet \ E < U, \ E \rightarrow \bullet \ U,
         U \rightarrow \bullet U + R, U \rightarrow \bullet R, R \rightarrow \bullet ! V, R \rightarrow \bullet V, V \rightarrow \bullet (E),
         V \rightarrow \bullet \text{ id}, V \rightarrow \bullet \text{ id}(M), V \rightarrow \bullet \text{ ent}, V \rightarrow \bullet \text{ cadena}
S_{51}=\{F \rightarrow function \ T1 \ id (\bullet \ A)\{C\}, \ A \rightarrow \bullet \ T \ id \ K, \ A \rightarrow \bullet, \ T \rightarrow \bullet \ int,
         T \rightarrow \bullet \text{ string}, T \rightarrow \bullet \text{ boolean}
S_{52} = \{S \rightarrow id \ L \ E \bullet ; , E \rightarrow E \bullet < U\}
S_{53} = \{S \rightarrow id(M) \bullet ; \}
S_{54}=\{M \rightarrow E Q \bullet \}
S_{55}=\{Q \rightarrow , \bullet E Q, E \rightarrow \bullet E < U, E \rightarrow \bullet U, 
         U \rightarrow \bullet U + R, U \rightarrow \bullet R, R \rightarrow \bullet ! V, R \rightarrow \bullet V, V \rightarrow \bullet (E),
         V \rightarrow \bullet \text{ id}, V \rightarrow \bullet \text{ id}(M), V \rightarrow \bullet \text{ ent}, V \rightarrow \bullet \text{ cadena}
S_{56} = \{S \rightarrow print(E) \bullet ;\}
S_{57} = \{S \rightarrow input(id) \bullet ;\}
S_{58} \hspace{-0.05cm} = \hspace{-0.05cm} \{S \rightarrow i\, f\, (E) \mid \bullet \ S1 \,, \ S1 \rightarrow \bullet \ \{S2\}G, \ S1 \rightarrow \bullet \ S, \ S \rightarrow \bullet \ i\, d \ L \ E; \,,
         S \rightarrow \bullet id(M);, S \rightarrow \bullet print(E);, S \rightarrow \bullet input(id);,
         S \ \rightarrow \ \bullet \ if (E) S1 \, , \ S \ \rightarrow \ \bullet \ return \ X \ ; \}
S_{59} = \{E \rightarrow E < U \bullet, U \rightarrow U \bullet + R\}
S_{60} = \{U \rightarrow U + R \bullet \}
S_{61} {=} \{ V \rightarrow (E) \quad \bullet \, \}
S_{62} = \{V \rightarrow id (M \bullet)\}
S_{63}=\{M \rightarrow E \bullet Q, E \rightarrow E \bullet < U, Q \rightarrow \bullet, Q \rightarrow \bullet, EQ\}
S_{64} = \{F \rightarrow function T1 id(A \bullet)\{C\}\}
S_{65}=\{A \rightarrow T \bullet id K\}
S_{66} = \{S \rightarrow id L E ; \bullet \}
S_{67} = \{S \rightarrow id (M); \bullet \}
S_{68}=\{Q \rightarrow E \bullet Q, E \rightarrow E \bullet C, Q \rightarrow \bullet, Q \rightarrow \bullet, EQ\}
S_{69} = \{S \rightarrow print(E); \bullet \}
S_{70} = \{S \rightarrow input(id); \bullet\}
S_{71}=\{S \rightarrow if(E) S1 \bullet \}
S_{72} = \{S1 \rightarrow \{\bullet \ S2\}G, \ S2 \rightarrow \bullet \ S \ S2, \ S2 \rightarrow \bullet \ S, \ S \rightarrow \bullet \ id \ L \ E;, \}
         S \ \rightarrow \ \bullet \ id \, (M) \, ; \, , \ S \ \rightarrow \ \bullet \ print \, (E) \, ; \, , \ S \ \rightarrow \ \bullet \ input \, (id \,) \, ; \, ,
         S \rightarrow \bullet \text{ if } (E)S1, S \rightarrow \bullet \text{ return } X ; 
S_{73} = \{S1 \rightarrow S \bullet \}
S_{74}=\{S2 \rightarrow S \bullet S2 , S2 \rightarrow S \bullet , S2 \rightarrow \bullet S S2, S2 \rightarrow \bullet S,
         \dot{S} \rightarrow ullet id \ L \ E \ ; , \ S \rightarrow ullet id \ (\ M \ ) \ ; , \ S \rightarrow ullet print \ (\ E \ ) \ ; ,
         S \rightarrow \bullet \text{ input (id)};, S \rightarrow \bullet \text{ if (E)} S1, S \rightarrow \bullet \text{ return } X;
```

```
S_{75}=\{F \rightarrow function T1 id (K) \bullet \{C\}\}
S_{76}=\{A \rightarrow T \text{ id } \bullet K , K \rightarrow \bullet, K \rightarrow \bullet , T \text{ id } K\}
S_{77}=\{Q \rightarrow , E Q \bullet \}
S_{78} = \{S1 \rightarrow \{S2 \bullet \} G\}
S_{79} = \{S2 \rightarrow S \bullet S2, S2 \rightarrow S \bullet, S2 \rightarrow \bullet S S2, S \rightarrow \bullet id L E;,
          S \rightarrow \bullet id(M); , S \rightarrow \bullet print(E); S \rightarrow \bullet if(E)S1; ,
           S \ \rightarrow \ \bullet \ input(id); , \ S \ \rightarrow \ return \ X; \}
S_{80} = \{F \rightarrow function T1 id (K) \{ \bullet C \}, C \rightarrow \bullet D C, C \rightarrow \bullet \}
          \mathrm{D} \ \rightarrow \ \bullet \ \mathrm{var} \ \mathrm{T} \ \mathrm{id} \ \ ; \, , \ \mathrm{S} \ \rightarrow \ \bullet \ \mathrm{id} \ \mathrm{L} \ \mathrm{E} \ \ , \ \mathrm{S} \ \rightarrow \ \bullet \ \mathrm{id} \ \ (\mathrm{M}) \, ; \, ,
           S \rightarrow \bullet \text{ print } (E);, S \rightarrow \bullet \text{ input } (id);, S \rightarrow \bullet \text{ if } (E) S1,
          S \rightarrow \bullet return X ; 
S_{81} = \{A \rightarrow T \text{ id } K \bullet \}
S_{82} = \{K \rightarrow, \bullet T \text{ id } K, T \rightarrow \bullet \text{ int }, T \rightarrow \bullet \text{ string }, T \rightarrow \bullet \text{ boolean}\}
S_{83} = \{S1 \rightarrow \{S2\} \bullet G , G \rightarrow \bullet \text{ else } \{S2\} , G \rightarrow \bullet \}
S_{84} = \{F \rightarrow function T1 id (K) \{C \bullet \}\}
S_{85}=\{C \rightarrow D \bullet C, C \rightarrow \bullet D C, C \rightarrow \bullet S C, C \rightarrow \bullet, D \rightarrow \bullet \text{ var T id } ;,
           S \rightarrow \bullet \text{ id } L E ;, S \rightarrow \bullet \text{ id } (M) ;, S \rightarrow \bullet \text{ print } (E) ;,
           S \rightarrow • input ( id ) ;, S \rightarrow • if ( E ) S1, S \rightarrow • return X ;}
S_{86}=\{C \rightarrow S \bullet C, C \rightarrow \bullet D C, C \rightarrow \bullet S C, C \rightarrow \bullet, D \rightarrow \bullet \text{ var T id } ;,
          S \rightarrow \bullet \text{ id } L E ;, S \rightarrow \bullet \text{ id } (M) ;, S \rightarrow \bullet \text{ print } (E) ;, S \rightarrow \bullet \text{ input } (\text{ id }) ;, S \rightarrow \bullet \text{ if } (E) S1, S \rightarrow \bullet \text{ return } X ;}
S_{87} = \{K \rightarrow , T \bullet id K\}
S_{88} = \{S1 \rightarrow \{S2\} G \bullet \}
S_{89} = \{G \rightarrow else \bullet \{S2\}\}\
S_{90} = \{F \rightarrow function T1 id (K) \{C\} \bullet \}
\begin{array}{l} S_{91} = \{K \rightarrow , \ T \ id \ \bullet K, \ K \rightarrow \bullet \ , \ K \rightarrow \bullet \ , \ T \ id \ K\} \\ S_{92} = \{G \rightarrow else \ \{ \bullet S2 \}, \ S2 \rightarrow \bullet \ S \ S2, \ S2 \rightarrow \bullet \ S, \ S \rightarrow \bullet \ id \ L \ E \ ;, \\ S \rightarrow \bullet \ id \ (M \ ) \ ; \ , \ S \rightarrow \bullet \ print \ (E \ ) \ ; \ , \ S \rightarrow \bullet \ input (id); \ , \end{array}
          S \rightarrow \bullet \text{ if (E)} S1, S \rightarrow \bullet \text{ return } X ;}
S_{93} = \{K \rightarrow , T \text{ id } K \bullet \}
S_{94} = \{G \rightarrow else \{ S2 \bullet \} \}
S_{95} = \{G \rightarrow else \{ S2 \} \bullet \}
S_{96} = \{C \rightarrow D C \bullet \}
S_{97} = \{C \rightarrow S \ C \bullet \}
S_{98} = \{D \rightarrow var \ T \ id ; \bullet \}
S_{99} = \{S2 \rightarrow S \ S2 \bullet \}
```

1.3 Conflictos y Tabla de Decisión

Como podemos observar en la tabla de decisión no hay ningún conflicto. Los posibles conflictos son:

Reducción-Reducción

Podríamos ver como en los posibles estados con este conflicto, en nuestro caso ninguno, se verifica que

 $\forall \{A \to \alpha \bullet, B \to \beta \bullet\} \subset S_x \Rightarrow \text{Follow}(A) \cap \text{Follow}(B) = \emptyset \text{ (Esto lo podemos observar al no tener dos entradas de reducción en la misma celda de cada fila de <math>S_x$)

Reducción-Desplazamiento

Podemos ver como en los posibles estados con este conflicto, S_0 , S_2 , S_3 , S_4 , S_6 , S_{11} , S_{22} , S_{28} , S_{29} , S_{34} , S_{41} , S_{50} , S_{51} , S_{59} , S_{63} , S_{68} , S_{76} , S_{79} , S_{80} , S_{83} , S_{85} , S_{86} , S_{91} , se verifica $\forall \{A \rightarrow \alpha \bullet b \gamma, C \rightarrow \beta \bullet\} \subset S_x \Rightarrow b \notin Follow(C)$ (Esto lo podemos observar al no tener una entrada de desplazamiento y otra de reducción en la misma celda de cada fila de S_x)

-	_	id entero cadena ()		= var int boolean s	int boolean string print input .	return function if else	\$ P D T F T A	K C S L M Q S' G X E U R V S"
				45	-	d11 d6	r49 1 2 3	4
S1	_	_	_	_	_	_		
S2	d7			d5	-	9p	r49 12 2 3	4
S3	d7	_ - -	_ _ _	d5	6p 8p	d11 d6 d10	r49 13 2 3	4
S4	d7			d5	6p 8p	d11 d6 d10	r49 14 2 3	
S5				d16 d18	d17		15	
86	r9	_ _ _	_ _ _	d16 d18	d17	_	20 19	
22	_	d22	d21 d23	123	_			
88		d24						
68	_ 	d25	 					
S10								
S11 r35	d34 d35	d36 d33	d31		_			
812	_	_			_		r1	
S13	_	_	_		_	_	r2	
814	_				_	_	r3	
S15	d37							
816	r5							
817	r6	_ 	_ _ _	_ _ _	_	_		
818	71	_	_	_	_	_		
819		- - -	- - -	- - -	- -	 		
S20	r10		_	_	_	_		
S21	r25 r25	r25 r25	r25					
S22	l_	929						
S23	l	r24	r24	- - -	- -	 		
S24	d34 d35	d36 d33	d31	- - -	_	_	 	
S25	d43							
826	d34 d35	d36 d33	d31	_ _ _	_			
S27 d45	_	_ _ _	 	_ _ _	_ _ _	_		
S28 r34	_ _ _	_ _ _			_ _	_		
S29 r37		r37	r37 d47 r37		r37	7		
S30 r39	_	r39	r39 r39 r39		r39	6		
S31	d34 d35	d36 d33						
S32 r41			r41 r41 r41		r41	1		
S33	d34 d35	d36	d31	_ _ _	_ _ _	_		
S34 r43	_	d50 r43 r43 r43	r43 r43	_ _ _ _	r43	3		
S35 r45		r45	r45 r45 r45	_ _ _ _	r45	2		
S36 r46		146	r46 r46 r46	·	146	9		
S37 d98								
838			_ _ _ _	_ _ _	_	_		
839	d34 d35	d36 d33	d31	_ _ _	_	_		
S40								
S41	_	r28		_ _ _	d55	5		54
S42	_	d56			_			
S43		d57						
S44	_	458			_ _ _	_		
S45		-	_ _ _	r23	r23 r23	r23 r23 r23	r23	
S46		d36		_ _ _	_	_	_	
S47	d34 d35	d36 d33	d31	_ _ _ _	_ _ _	_ _ _		
S48 r40	_		r40	_ _ _ _	r40			
849				_	_	_		

	{ } id entero cadena	la () +	 V	= var int	boolean string print input	print inpu	t . return	rn function	n if else	se \$ P	D T	F T' A	K C S	T M O	S G	X E U	R V S"
820		d33 r27	d31		-1-	_			_	_	- -	-		62		63 29	32
S21		112	_	d16	3 d18 d17	_	_	_	_	_	65	64	_	_		_	
S52 d66	_ _ _	_ _ _	d46	_	_	_	_	_	_ _	_ _	_	_	_	_	_	_ _	
S53 d67									- - -	<u>-</u>	_ - -	_		_			_
S54		r26	757	_ -				_ -	- -	= - - -						06 89	- 68 08
S56 d69	- -	_ _	Ten	- - -	- -	_	- -	-	- - - -	= = - -	- - - - -	 - -	- -	- - - -		64	
S57 d70	- - - - -			-	- - -			-	-	= -							
	d72 d7			_		6p 8p	d11		010	_	- -		73	_	71		-
S59 r36	_ _ _	r36 d47 r36	r36	_	_	_	136	_	_ _	_ _	_ _ _	_	_	_	_	_	_
S60 r38	_	r38 r38 r38	r38	_	_	_	r38	_	_	_	_	_		_		_	_
S61 r42		r42 r42 r42	r42	_	_	_	r42		_	=	_	_		_			_
S62	_	d74	_		_		_	_	_	_	_ _ _	_	_		_	_	
Se3	_ _ _	128		_	_	_	d55	_	_	_	_ _ _	_	_	54	_	_	_
S64	_ _ _	d75		_	_	_	_		_	= -	_	_	_	_		_ _ _	_
S65	d76	_ _ _ _		- -	_ _ _	_	- -	-	- -	<u>-</u>	_ _ _	_ _ _	_ _ _	_	_	_ _ _	_
998	118 118	_ - _ -		118	_ - _ -		_ - _ -	_ -	- -	r18	_ - _ -			_ -	_		_
1 201	119 119			- III	- - - -	ris ris	-1-	a l ris	1119	= I	_ -	_ -	_ -	_ -	- - - -	- - - -	_ -
898		r28			_ - _ -	- -	d22	_ -	_ -			_ -	_ - _ -				
698	r20 r20	_ - _ -	_ - _ -	r20	_ -		_ _ .	- -	_	r20	_ _ :	_	_ .	_ _ :		_ _ - _ -	_
S70	r21 r21			r21			_	_	r21	r21		_ _ _	_				
128	r22 r22			r22		-	_	2 r22	r22	r22	_ _ _	_	_ _ 	_	_ _ _	_ _ _	_
S72	d7	_ _ _ _	_ _ _	- -	_ _ _	-1	_	-	q10	<u>-</u>	_ _ _	_	79	_	_	_ _ _	_
S73	r31 r31			r31	_	r31 r31	-	1 r31	r31	r31		_		_			
r44		r44 r44	r44	_		_	r44	_	_ _	<u>-</u>	_ _ _	_ _ _	_	_		_ _ _	
	d80			- - - -	_ - _ -	_		_ -	- - - -	= - - -		_		_ - _ -			
928	_ -	113		- - - -	_ -	_ -	d82	_	- - - -	- - - -		_ -	81				
S72 - 228		r29		- - - -		_	_ -	_ -	- - - -	= = - -	_ - - -	_ -	_ -	_ -		= - - -	
SZ9	r48 d7			-	- - -	6p 8p	d11	-	q10	_			62	-			- 66
088	r17 d7			- d5		-	_	1	q10	_	85		84 86				
881		r11		_		-	_	_	-	_	- - -			_			_
882		_	_	d16	3 d18 d17	_	_	_	_ _	- -	87	_ _ _	_ 	_	_ _ _	_ _	_
883	r33 r33	_ _ _		r33	_	r33 r33	r33	3 r33	r33 d8	r33 d89 r33	_	_	_	_		_	
884	d90	_ _ _	_ _ _	_ _	_	-	- -	_	_ _	<u>=</u>	_	_		_	_	_ _ _	_
S85		_ _ _ _	_ _ _	d5		-	_		d10	_ _	_ _ _	_ _ _	_	_	_ _ _	_ _ _	_
988	r17 d7			- d5	_ -	6p 8p	d11	_ -	q10	<u>-</u> -	82	_ _ _ _	98 26	_ - _ -		_ _ _	_
2887	d91	_ - _ -			_ -	- -	- - - -	- -				_ -	- - - -				
	r30 r30	_ - _ -		130 I	_ - _ -	r30 r30	130	n r30	130	130 I		_ -					
	d92					_ -		- -			_ - _ - _ -		_ - _ -				
1068			_ - - -	2i -	- - - -	81 - 81 - 82 - 83 - 83 - 83 - 83 - 83 - 83 - 83	21 	Σ <u>ι</u> -	- SI -	- I	- - - -	_ _ _ _		_ -	_ -	_ _ _ _	-
168		- r13		- - - -	_ -	- -	- d85 - d85	_	- - - -	- - - -		_ - _ -		_ -			_ -
892	d7			_	_	6p 8p	d11	_	q10	<u>-</u>		_	62				- 34
893		r14		_			_ -	_	_ _	<u>-</u>		_ _ _	_				
894	d95	_ _ _ _		- -	_ _ _	_	- -	_	_ _	<u>-</u>	_ _ _	_ _ _	_ _	_	_ _ _	_ _ _	_
895	r32 r32			r32		r32 r32	r32	2 r32	r32	r32		_				_ _ _	
968	r15			_	_		_	_	_ _	<u>-</u>		_	_				
Z6S			_ - _ -	- - - -			_ - _ -	_	- - - -	<u>-</u> -	- -		_ - -	_ - _ -			
868	r4 r4			r4	_	r4 r4	- r4	- r4	r4 -	-14 		_	_ .	_ - _ -		_ _ - _ -	_
868	147	_			_		_		_	= -							

1.4 Errores

En las celdas vacías de cada fila se lanzan los siguientes errores:

S₀, S₄, S₇, S₁₄, S₈₅, S₈₆, S₉₆, S₉₇: Error 1: "Sentencia no válida"

S₁: Error -1 : "No se pudo derivar la raíz"

S₂, S₅, S₁₂, S₁₅, S₃₇, S₉₈: Error 2: "Declaración incorrecta de variable"

 S_3 , S_6 , S_{13} , S_{19} , S_{38} , S_{51} , S_{64} , S_{65} , S_{75} , S_{76} , S_{80} , S_{81} , S_{82} , S_{84} , S_{87} , S_{90} , S_{91} , S_{93} : Error 3: "Declaración incorrecta de función"

 $S_8,\,S_{24},\,S_{42},\,S_{56},\,{}_{S69}\!\!:$ Error 4: "Sentencia print incorrecta"

S₉, S₂₅, S₄₃, S₅₇, S₇₀: Error 5: "Sentencia input incorrecta"

S₁₀, S₂₆, S₄₄, S₅₈, S₇₁, S₇₃: Error 6: "Sentencia condicional simple incorrecta"

 $S_{11},\,S_{27},\,S_{28},\,S_{45}\colon$ Error 7: "Sentencia return incorrecta"

 S_{16} , S_{17} , S_{18} , S_{20} : Error 8: "Tipo incorrecto"

 S_{21} , S_{23} , S_{39} , S_{52} , S_{66} : Error 9: "Asignación incorrecta"

 S_{22} , S_{40} , S_{41} , S_{53} , S_{54} , S_{55} , S_{63} , S_{67} , S_{68} , S_{77} : Error 10: "Llamada a función incorrecta"

 $S_{29}, S_{30}, S_{31}, S_{32}, S_{33}, S_{34}, S_{35}, S_{36}, S_{46}, S_{47}, S_{48}, S_{49}, S_{50}, S_{59}, S_{60}, S_{61}, S_{62}, S_{74}$: Error 11: "Expresión incorrecta"

 S_{72} , S_{78} , S_{79} , S_{83} , S_{88} , S_{89} , S_{92} , S_{94} , S_{95} , S_{99} : Error 12 "Sentencia condicional compuesta incorrecta"

2 Anexo de Pruebas

Error 1:

```
1 var int a;
2 var b;
3 a = 3;
4 b = a;
5 if (a < b) b = 1;
6 if (b < a) b = 8;
7 a = a + b;
8 print (a);
9 print (b);</pre>
```

Error 2:

```
var string texto;
function pideTexto ()

{
   print ('Introduce un texto');
   input (texto);

}

function imprime (string msg,)

{
   print (msg);

}

pideTexto();

var string textoAux;

textoAux = texto;

imprime (textoAux);
```

> Error Sintactico: Declaracion incorrecta de funcion. Linea: 7

Error 3:

```
var int a;
var int b;
a = 3;
b = a;
var boolean c;
c = a < b;
if (c) {
b = 1;
} else {
c = b < a;
if (c) b = 4;
print (a);
print (b);</pre>
```

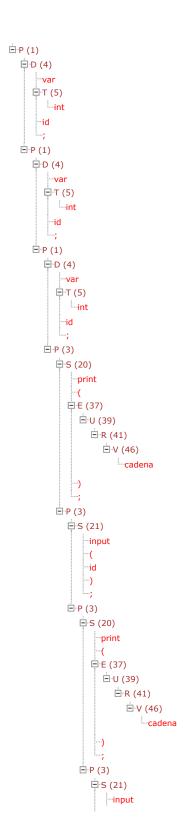
> Error Sintactico: Sentencia condicional compuesta incorrecta. Linea: 14

Prueba 1 Correcta:

```
1 var int a;
2 var int b;
3 var int c;
4 print ('Introduce el primer operando');
5 input (a);
  print ('Introduce el segundo operando');
7
  input (b);
8 function int suma (int num1, int num2)
9 {
10
  var int res;
11
    res = num1+num2;
12
    return res;
13 }
14 c = suma (a, b);
15 print (c);
```

Parse a Derechas:

Árbol sintáctico:



```
···(
···id
  ···)
<u></u>-F (8)
   function
    T1 (10)
     ⊟ T (5)
       int
     ∵id
     ...(
    A (11)
      □-T (5)
—int
—id
□-K (14)
       □ T (5)
int
        Ē K (13)
         lambda
      ····)
    C (15)
      Ē ·D (4)
        var
□ T (5)
□ int
□ id
□ ';
       E C (16)
         - S (18)
          Ė U (38)
                .
□ U (39)
                 Ē-R (41)
                   _ v (43),
—id
—+
                Ē-R (41)
                  ÷...∨ (43)
                    id
         Ė C (16)
            Ē S (23)
            return
             □·X (34)
□·E (37)
```

```
.
⊡. U (39)
                    Ē ·R (41)
                      ...V (43)
                         id
          Ē C (17)
             lambda
Ē ··P (3)
  Ė S (18)
    Ė (37)
       Ū U (39)
          .
∃∵R (41)
            ÷...∨ (44)
               ···id
               —(
□-M (26)
                 Ē (37)
                   Ė U (39)
                     .
∃ ··R (41)
                       Ē·V (43)
                            i...id
                 Ē-Q (29)
                    ...,
-E (37)
                      .
∃.·U (39)
                        Ė ·R (41)
                           ÷ ∨ (43)
                              i...id
                    Q (28)
                       lambda
  Ē-P (3)
     Ė S (20)
       print
(
= E (37)
        <u>-</u>U (39)
          .
∃∵R (41)
               ÷ ∨ (43)
                 i...id
     Ė P (49)
        lambda
```

Prueba 2 Correcta:

```
var string texto;
function imprime (string msg)

{
   print ('Mensage introducido:');
   print (msg);
}

function pideTexto ()

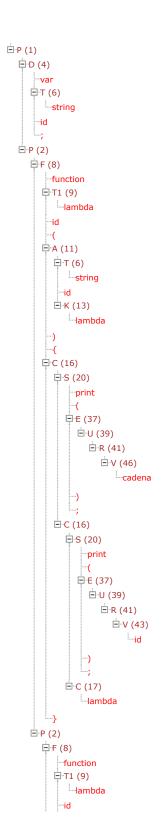
{
   print ('Introduce un texto');
   input (texto);

}

pideTexto();
imprime (texto);
```

Parse a Derechas:

Árbol sintáctico:



```
....(
□··A (12)
  lambda
 Ē-S (20)
    Ė-U (39)
        Ė ·R (41)
          Ė ∨ (46)
             cadena
      ···)
;
    Ē ·C (16)
     Ġ S (21)
      --input
---(
---id
     lambda
Ē ··P (3)
  Ė ·S (19)
  id (
   ĒM (27)
    lambda
    ···)
;
  Ē-P (3)
    Ë S (19)
     id ....(
-...(
-...(26)
        Ė-U (39)
          Ė R (41)
          ⊟ V (43)
       Ē Q (28)
        lambda
    Ē P (49)
      lambda
```

Prueba 3 Correcta:

```
1 var string s;
2
   var int
              uno:
3 var int
              UNO;
4 function int Factorial (int n)
6
    if (n < 0) return 1;
7
     return n + Factorial (n + 1);
8 }
9 var int For;
10 var int functional;
11 var int While;
13 function imprime (string s, string msg, int f)
14 {
    print (s); print (msg); print (f);
15
16
    return;
17 }
18 function string cadena (boolean log)
19 {
20
     if (!log)
21
     {
22
       imprime (s, 'hola', 33);
23
       if (uno < UNO) return s;</pre>
24
     }
25
    else
26
    {
27
       return 'Fin';
28
     }
29 }
30 s = 'El factorial ';
31
32 print (s);
33 print ('Introduce un numero.');
34 input (num);
35 var
36 boolean
37 booleano;
38 if (num < 0)
                  print ('No existe el factorial de un negativo.');
39 imprime (cadena (booleano), 'recursivo es: ', Factorial (num));
```

Parse a Derechas:

Árbol sintáctico:

```
.
□ P (1)
   .
□ D (4)
    var

T-T (6)

string

id

;
   Ē P (1)
     ⊡ D (4)
        -var
-T (5)
-int
-id
      .
□ P (1)
         → D (4)

→ var

→ T (5)

→ int

→ id

→ ;
         .
□ ·D (4)
         Ē ··P (2)
            ≒ F (8)
               function
               <sup>‡</sup>.⊤1 (10)
                Ė⊤T (5)
                   int
                --(
               A (11)
                 ⊟ T (5)
int
id
⊟ K (13)
                   lambda
               Ċ (16)
                  ÷ S (22)
                    if
(
= E (36)
                       Ē-E (37)
                         Ē-U (39)
                            .
∃∵R (41)
                                ....V (43)
                                   i...id
                        ---<
⊡--U (39)
                           .
∃ ·R (41)
                              ÷.∨ (45)
                                 ent
```

```
Ė S1 (31)
        ÷ S (23)
          return
          □ X (34)
            .
E (37)
             Ė ·U (39)
               Ē-R (41)
                 Ė∵V (45)
                     ent
    □ S (23)
        return X (34)
         Ē-E (37)
            Ė ·U (38)
              Ū·U (39)
               Ē-R (41)
                 ∃ V (43)
              Ē ··R (41)
                 id
(
                   □ M (26)
                    <u>≒</u>..E (37)
                      .
∃.·U (38)
                         Ū·U (39)
                          Ē R (41)
                           id
                         □ R (41)
                         Ė∵V (45)
                            ent
                     Ē Q (28)
                       lambda
      ⊡ C (17)
        lambda
.
□ P (1)
 □ D (4)
   var

T (5)
    id
;
  Ē ··P (1)
```

```
□ D (4)
  var (5)
  int
id
 □ D (4)

var
□ T (5)

int

id

;
Ē-P (1)
  Ē P (2)
    ₽ F (8)
     function
       Ё∙Т1 (9)
         lambda
         ...id
       --(
---A (11)
         T (6)
string
         ⊞ K (14)
           □ T (6)
           string
            Ē K (14)
             T (5)
              ⊟ K (13)
               lambda
       □ C (16)
         .

⇒ S (20)
          print
(
E-E (37)
E-U (39)
              Ė ·R (41)
                Ė V (43)
                     id
             ···)
         Ē ·C (16)
            ÷ S (20)
             print (
```

```
Ė (37)
          .
∃ · U (39)
           .
∃∵R (41)
             id
        ····)
      Ē ·C (16)
        Ë ·S (20)
         ---print
          E (37)
          ⊡.·U (39)
            .
∃∵R (41)
              ⊟.·V (43)
          ···)
;
        Ē-C (16)
          □ S (23)
           return
            ± X (35)
            lambda
          Ē C (17)
            lambda
  ····}
Ë P (2)
  .
- F (8)
   function
   ⊤T1 (10)
    ⊟ T (6)
      string
     ···id
    --(
    Ā (11)
     ⊟⊤ (7)
—boolean
     ⊟ K (13)
      lambda
    --)
--{
    □·C (16)
     Ė∙S (22)
      Ė ·R (40)
             □ V (43)
```

```
i...id
--)
--S1 (30)
 Ģ S (19)
    id
(
E-M (26)
      ₽-E (37)
        Ė-U (39)
          ⊟ R (41)
            ÷ ∨ (43)
             id
        Ė Q (29)
         ⊟ E (37)
⊟ U (39)
           .
∃∵R (41)
           UV (46)
         Ė Q (29)
           □ E (37)
            Ū U (39)
             Ë R (41)
              ent
           Ē Q (28)
              lambda
      ···)
    = S2 (48)
     .
∃..S (22)
       if
--(
--E (36)
        Ė-E (37)
          Ē-U (39)
           ⊟ R (41)
             Ė·V (43)
               ---id
          ····<
         .
∃ U (39)
          .
∃ R (41)
           Ė V (43)
            id
       =-)
=-S1 (31)
        return
```

```
Ė X (34)
                           .
∃. E (37)
                             .
∃.U (39)
                                .
□ R (41)
                                  ÷...∨ (43)
                                      id
            ---}
          Ġ (32)
              --else
---{
             = S2 (48)
                ÷ S (23)
                   return
EX (34)
                     ÷ E (37)
                        .
∃ · U (39)
                           .:
-R (41)
                             ÷ ∨ (46)
                                 cadena
     .
∃...C (17)
        lambda
   ····}
.
∃ · P (3)
  Ģ S (18)
     id
-L (25)
     Ē E (37)
       .
∃.·U (39)
          .
∃ R (41)
             ÷...∨ (46)
                cadena
  .
□ P (3)
     Ė S (20)
       print
--(
--E (37)
         Ė-U (39)
            Ē R (41)
               Ē V (43)
                  id
        ···)
     Ē ·P (3)
        Ġ (20)
         --print
```

```
Ė (37)
   Ū · U (39)
    .
⊟ R (41)
       Ů·V (46)
         cadena
.

⇒ S (21)
   input
(
id
;
  Ē P (1)
     .
□ D (4)
    -P (3)
      ÷ S (22)
       if
(
= E (36)
         Ë-E (37)
           Ė U (39)
             .
∃∵R (41)
               ÷ ∨ (43)
           --<
          Ė-U (39)
           Ë R (41)
            ...V (45)
              ent
        ---)
----S1 (31)
          ÷ S (20)
           print
(
= E (37)
             Ė U (39)
               Ė R (41)
                 cadena
      Ē ··P (3)
        ÷ S (19)
         id
...(
```

```
∺ M (26)
    Ė (37)
      .
⊟ U (39)
        .
⊟ R (41)
          ÷ ∨ (44)
             id
(
= M (26)
              Ë E (37)
                 Ů··U (39)
                   .
∃ ·R (41)
                    ÷...∨ (43)
                        i...id
               Ė Q (28)
                  lambda
    Ė ·Q (29)
      E (37)
        .
∃ U (39)
         Ė∵R (41)
           Ė V (46)
               cadena
      Ū · Q (29)
        .
⊟ U (39)
           .
∃ ··R (41)
               ÷ ∨ (44)
                 id (
                 ⊢M (26)
                   Ë E (37)
                     Ė ··U (39)
                       Ė ··R (41)
                         ÷...∨ (43)
                             id
                   Ē ·Q (28)
                      lambda
        Q (28)
           lambda
lambda
```