# 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	Disc	eño del Analizador Sintáctico	2
	1.1	Gramática	2
	1.2	Autómata Reconocedor de Prefijos Viables	3
		1.2.1 Estados del autómata	4
	1.3	Conflictos	6
<b>2</b>	Ane	exo de Pruebas	9

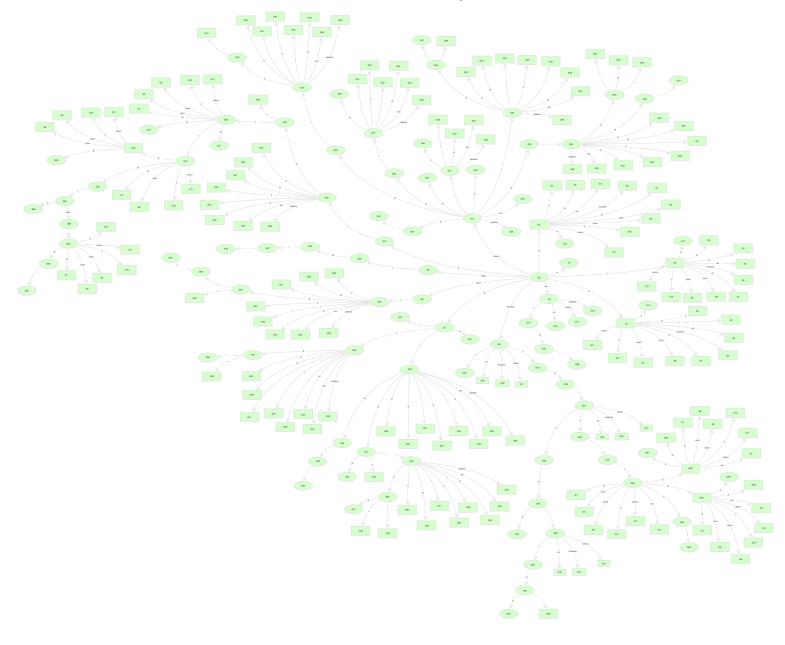
### 1 Diseño del Analizador Sintáctico

#### 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 \text{ 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 \}
         D \rightarrow \bullet \text{ var } T \text{ 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 } (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 • id L E ;, S \rightarrow • id ( M ) ;, S \rightarrow • print ( E ) ;,
          S \rightarrow \bullet input ( id ) ;, S \rightarrow \bullet if ( E ) S1, S \rightarrow \bullet 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 \ \text{id} \ \bullet K, \ K \rightarrow \bullet \ , \ K \rightarrow \bullet \ , \ T \ \text{id} \ K\} \\ S_{92} = \{G \rightarrow \text{else} \ \{ \ \bullet \ S2 \}, \ S2 \rightarrow \bullet \ S \ S2 \ , \ S2 \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}); \ , \end{array}
         S \rightarrow • if ( E ) S1, S \rightarrow • 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

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

#### Reducción-Reducción

Podemos ver como en los posibles estados con este conflicto, S...., se verifica que  $\forall \{A \to \alpha \bullet, B \to \beta \bullet\} \subset S_x \Rightarrow \text{Follow}(A) \cap \text{Follow}(B) = \emptyset$  (Esto lo podemos observar al no tener dos entradas de reducción en la misma celda de cada fila de  $S_x$ )

#### Reducción-Desplazamiento

Podemos ver como en los posibles estados con este conflicto, S...., se verifica que  $\forall \{A \to \alpha \bullet b \gamma, C \to \beta \bullet\} \subset S_x \Rightarrow b \notin Follow(C)$  (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   enterc	entero   cadena   (   )	= 	=   var	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"
os	d7		-	- GP		d11   d6   d10	r49   1   2     3	4
S1							в	
S2	47			d5		- qe	: =	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	4
S5	_	_ _ _	_	d16   d18	8   d17		15	
Se	61	_ _ _	_	d16   d18	8   d17		20   19	
S7	_	d22	d21	d23	_ _ _ _	_		
88	_ _ _	d24	_ _ _	_ _ _	_ _ _ _	  		
6S	_	d25	_					
S10	_	d26	_					
S11   r35	d34   d35	d36   d33		_ _ _				
S12	_	_	_	_ _ _	_	_	11	
S13	_	_	_	_	_	_	r2	
814				_			r3	
S15	d37							
S16	r5							
S17	lu6	_	_	_	_	_		
818	Tr	_	_	_				
819	829	-  -  -				_		
S20	r10	-  -  -		_		_		
S21	r25 r25	r25 r25	r25	_				
S22	l_	989						
S23	l	r24	r24			_		
S24	d34   d35	d36		_ _ _	_	_		
S25	d43	_ _ _			_ _ _			
826	d34   d35	d36   d33	d31	_ _ _	_ _ _	_ _ _		
S27   d45	_ _ _	_ _ _ _	_ _ _	_ _ _	_ _ _ _	_ _ _ _		
S28   r34	_	_ _ _	d46	_	_ _ _ _	_	_ _ _ _	
S29   r37	_ _ _	137	r37   d47   r37	_ _ _		r37		
S30   r39	_	139	r39   r39   r39	_		r39	   	
831	d34   d35	d36   d33	_ _ _	_	_ _ _ _	_	_ _ _ _ _	
S32   r41	_	_	r41   r41   r41	_		r41	_ _ _ _ _	
S33	d34   d35	d36					_ _ _ _	
S34   r43	_	d50   r43	d50   r43   r43   r43	_		r43	_ _ _ _ _	
S35   r45	_	r45	r45   r45   r45	_		r45	_ _ _ _ _	
S36   r46		146	r46   r46   r46		_	r46	_ _ _ _ _ _	
S37   d98	_ _ _	-  -  -	_  _  _		_  _  _	_  _  _	_ _ _ _ _ _	
838		_	_  _  _			_  _  _	_ _ _ _ _	_
839	d34   d35	d36   d33	d31	_	_ _ _ _	_ _ _ _	_ _ _ _ _	
S40	_ _ _	d53	_ _ _	_ _ _	_ _ _ _	_ _ _		
S41		r28			p	d55	_ _ _ _	
S42	_		d46	_	_	_		
S43		d57						
S44		82P       128	446					
S45			_	r23	r23   r23	r23   r23   r23	r23	
S46		9EP				_ _ _ _	_ _ _ _ _	
847	d34   d35	d36   d33						
S48   r40		1.40	r40			r40		
S49		d61						

	{   }   id   entero   cadena	_ =   ;   !   +   (   )	-=   var   int   boolean   string	;   print   input   ,	return   fu	function   if	else   \$    P	DITFT	A   K   C   S   L	M   Q   S'   G   X   E   U	U   R   V   S"
S20	d35 d36	d33   r27     d31						_	_		윤
S51		r12	d16   d18   d17	_	_	_	_	65	64		
S52   d66	_		_	_	_	_	_		_ _ _		_ _ _
S53   d67				_ _ _	_		_				
S54		r26	-		_		_		—   - —   - —   -		-
S55	d34   d35   d36	d33       d31	_   -		_ - _ -	_ -	_				29   30   32
S26   d69			_   -	_ -	_ -	_	_	_ - - -	-  -  -		
d70					_ -						
	d72   d7		_	6p   8p	d11	d10	_		73	71	_ _ _
S59 r36		r36   d47   r36	_	136		_	_	_	_ _ _	_ _ _ _	_ _ _
S60   r38	_ _ _	r38   r38	_ _ _	r38		_	_	_ _ _	_ _ _		_ _ _
S61   r42		r42   r42   r42		r42		_	_				_ _ _
S62	_	d74		_	_		_	_ _ _	_ _ _ _	_ _ _ _	_ _ _
893		r28     d46	_	d55		_	_	_ _ _	_ _ _ _	54	  -  -
864	_ _ _ _	d75	_ _ _	_ _ _	_		_	_ _ _	_ _ _ _	_ _ _ _	_ _ _
S65	d76				_	_	_				_ _ _
998	r18   r18	  	r18	r18   r18	r18	r18   r18	118	_ _ _			_ _ _
298	r19   r19		r19	r19   r19	r19	r19 r19	r19	_ _ _	_ _ _ _	  	_ _ _
898	_ _ _	r28     d46	_	d55		_	_	_ _ _	_ _ _		_ _ _
69S	r20   r20		r20	-	-		r20	_ _ _	_ _ _ _	  	_ _ _ _
870	r21   r21		r21	_	_	_	r21				
S71	r22   r22	  	r22	-	-	r22   r22	r22	_ _ _			_ _ _
S72	d7	  	_ _ _	6p   8p	d11	d10	_	_ _ _	62		78
S73	r31   r31		r31	r31   r31	r31	r31 r31	r31				_ _ _
r44		r44   r44   r44	_	r44		_	_	_ _ _	_ _ _ _	_ _ _ _ _	_ _ _ _
	d80           08p			_	_	_	_				
S76		r13		d82					81		
S77		r29	_ - _ - _ -	_ -	_ - _ .		_	_ - - - - -			
878	d83	-  -  -	_   - _   - _   -		_		_			-   -   -	
879	r48 d7			- -	- IIp	q10	_		1   79   1	- - - - - -	66
280	r17   d7		dp	- dg - dg -	d11	qTo	_ -	68	84 80		
188				- - - -	- - - -		= = - -		- - - - - -		- - - -
202			on on on		- -	- -	=   oot	70 -	-		_ -  -  -  -
2000				cel   cel	eel –	cel cel	cer   con   cer	- - - -	-		- - - -
200 S				- op - sp		910	= = - -		98 96		
988	-1-			- -	911	910	=		98 26	-	- - - - -
S 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2				- -	- - -						
888	130   130		r30	r30   r30	r30	r30   r30	r30				
	d92			- -	- -	1-	-				
	1 r8		r s	r8 r8	- R1	r8 r8	- N		-   -   -		
S91		r13		-	. _	-	_	_	93		  -  -
892	d7			6p   8p	d11	d10	_		62		
893		r14			-		_	_			
894				_	_		_	_			_
S95	r32   r32		r32	r32   r32	r32	r32 r32	r32	_			
968	r15		_	_	_	_	_				
268	r16		_	_	_	_	_	_ _ _	_ _ _ _	_ _ _ _	_ _ _ _
868	r4   r4		r4	r4   r4	r4	r4 r4	r4	_ _ _	_ _ _ _		_ _ _ _
899	r47		_ _ _ _	_	_	_	<u>=</u>	_ _ _	_ _ _ _	_ _ _ _	_ _ _

### 2 Anexo de Pruebas

#### 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);
```

A 5 4 5 4 5 4 46 41 39 37 20 21 46 41 39 37 20 21 5 10 5 5 13 14 11 5 4 25 43 41 39 43 41 38 37 18 43 41 39 37 34 23 17 16 16 15 8 25 43 41 39 37 43 41 39 37 28 29 26 44 41 39 37 18 43 41 39 37 20 49 3 3 2 3 3 3 3 1 1 1

#### Prueba 2 Correcta

```
1 var string texto;
2 function imprime (string msg)
3 {
4
    print ('Mensage introducido:');
5
    print (msg);
6 }
7 function pideTexto ()
8
    print ('Introduce un texto');
9
10
     input (texto);
11 }
12 pideTexto();
13 imprime (texto);
```

A 6 4 9 6 13 11 46 41 39 37 20 43 41 39 37 20 17 16 16 8 9 12 46 41 39 37 20 21 17 16 16 8 27 19 43 41 39 37 28 26 19 49 3 3 2 2 1

#### Prueba 3 Correcta

```
1 var string s;
   var int
              uno;
3 var int
              UNO;
4 function int Factorial (int n)
6
    if (n < 0) return 1;</pre>
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);
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));
```

#### 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 Sintactico: Declaracion incorrecta de variable. Linea: 2

#### Error 2

```
var string texto;
function pideTexto ()

function pideTexto ()

function imprime (string msg,)

function imprime (string msg,)

{
```

```
9  print (msg);
10 }
11 pideTexto();
12 var string textoAux;
13 textoAux = texto;
14 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