

# 1<sup>er</sup> PARCIAL LENGUAJES DE PROGRAMACIÓN Y PROCESADORES DE LENGUAJES. 2020-21

## PARTE I: ANÁLISIS SINTÁCTICO DESCENDENTE

1. Dada la siguiente gramática:

- 1:  $S \rightarrow x A B C$
- 2:  $A \rightarrow z B y$
- 3:  $A \rightarrow S y B m$
- 4:  $A \rightarrow \varepsilon$
- 5:  $B \rightarrow v B C y$
- 6:  $B \rightarrow \varepsilon$
- 7:  $C \rightarrow m$
- 8:  $C \rightarrow \varepsilon$

a) (0.75 pto) Escribe en el cuadro de texto adjunto los conjuntos de SIG(S), SIG(A), SIG(B) y SIG(C).

SIG(S) = { \$, y }  
 SIG(A) = { v, m, \$, y }  
 SIG(B) = { m, \$, y }  
 SIG(C) = { \$, y }

b) (1 pto.) Construye la tabla de análisis LL(1). En el cuadro de respuestas indica las acciones, distintas de error, correspondientes a todos los no-terminales.

	x	y	z	v	m	\$
S	(1, xABC)					
A	(3, SyBm )	(4, $\varepsilon$ )	(2, zBy)	(4, $\varepsilon$ )	(4, $\varepsilon$ )	(4, $\varepsilon$ )
B		(6, $\varepsilon$ )		(5, vBCy)	(6, $\varepsilon$ )	(6, $\varepsilon$ )
C		(8, $\varepsilon$ )			(7, m)	(8, $\varepsilon$ )

c) (1 pto.) A partir de la tabla de análisis LL(1) calculada anteriormente, realiza paso a paso la traza de análisis LL(1) para la cadena: x z v m y y m

(S, xzvmymm\$, )	-(xABC, xzvmymm\$, 1 )	-(ABC, zvmymm\$, 1 )
-(zByBC, zvmymm\$, 1-2 )	-(ByBC, vmymm\$, 1-2 )	-(vBCyyBC, vmymm\$, 1-2-5 )
-(BCyyBC, myym\$, 1-2-5 )	-(CyyBC, myym\$, 1-2-5-6 )	-(myyBC, myym\$, 1-2-5-6 -7)
-(yyBC, yym\$, 1-2-5-6 -7)	-(yBC, ym\$, 1-2-5-6 -7)	-(BC, m\$, 1-2-5-6 -7)
-(C, m\$, 1-2-5-6 -7-6)	-(m, m\$, 1-2-5-6 -7-6-7)	-( , \$, 1-2-5-6 -7-6-7)

d) (0.25 pto) A la vista de la tabla de análisis, ¿Es una gramática LL(1)? Justifica tu respuesta

Si que es LL(1) porque no hay ninguna celda con más de una acción (derivación).

e) (0.5 pto) Tras aplicar el esquema de eliminación de recursión a izquierdas a la gramática

$A \rightarrow A x A$   
 $A \rightarrow A z$   
 $A \rightarrow z x$   
 $A \rightarrow \varepsilon$

la gramática resultante es:

a) $A \rightarrow z x A'$ $A \rightarrow A'$ $A' \rightarrow x A A'$ $A' \rightarrow z A'$ $A' \rightarrow \varepsilon$	b) $A \rightarrow x A A'$ $A \rightarrow z A'$ $A' \rightarrow z x$ $A' \rightarrow \varepsilon$	c) $A \rightarrow x A A'$ $A \rightarrow z A'$ $A' \rightarrow z x$	d) $A \rightarrow z x A'$ $A \rightarrow A'$ $A' \rightarrow x A A'$ $A' \rightarrow z A'$
--	--	--	--

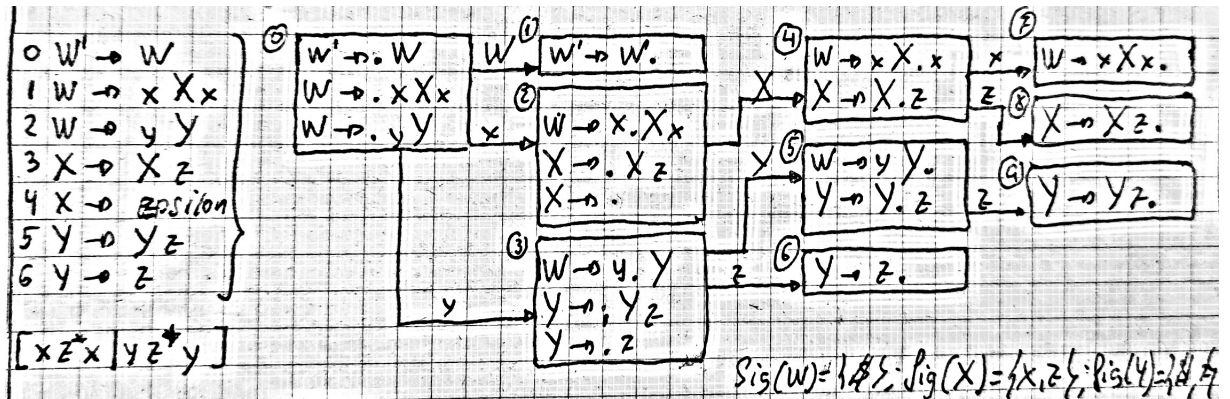
La solución correcta es la A

## PARTES II y III: ANÁLISIS SINTÁCTICO DESCENDENTE Y ANÁLISIS SEMÁNTICO

1. Dada la siguiente gramática:

$$\begin{array}{lll} W \rightarrow x X x & X \rightarrow X z & Y \rightarrow Y z \\ W \rightarrow y Y & X \rightarrow \epsilon & Y \rightarrow z \end{array}$$

- a) (1,5 ptos.) Construid la colección canónica de conjuntos de ítem LR(0). Proporcionad el conjunto de ítem válidos para los siguientes prefijos viables:  $\epsilon$   $x X$ .



- b) (1 pto.) A partir de la colección canónica de conjuntos de ítem LR(0), construid la tabla de análisis SLR(1).
- c) (1 pto.) Empleando la tabla de análisis SLR(1), calculada anteriormente, proporcionad la traza de análisis SLR(1) para la cadena:  $x z x$

	x	y	z	\$	W	X	Y
0	D-2	D-3			1		
1				ACC			
2	R-4		R-4		4		
3			D-6			5	
4	D-7		D-8				
5			D-9	R-2			
6			R-6	R-6			
7				R-1			
8	R-3		R-3				
9			R-5	R-5			

	x	z	x\$	—
0	x 2	z x\$	4	
0 x 2	X 4	z x\$	4	
0 x 2 X 4	z 8	x\$	4 3	
0 x 2 X 4		x\$	4 3	
0 x 2 X 4 x 7		\$	4 3 1	
0 W 1		\$	4 3 1	

- d) (0.5 ptos.) Justificad brevemente las respuestas:

- 1) Dada la colección canónica de conjuntos de ítem LR(0), ¿es una gramática LR(0)?
- 2) A la vista de la tabla de análisis SLR(1), ¿es una gramática SLR(1)?

**No** es LR(0) ya que existen estados con desplazamientos y reducciones. **Si** es SLR(1) ya que la tabla de análisis SLR(1) no tiene entradas múltiples.

- e) (0.5 ptos.) Al construir la colección canónica de conjuntos de elementos LR(0) hemos obtenido el estado:  $\{ [A \rightarrow A \text{ and } A.], [A \rightarrow A. + A], [A \rightarrow A. \text{ and } A] \}$

Si sabemos que  $SIG(A) = \{ +, \text{and} \}$  y que la precedencia del operador “+” es mayor que la del operador “and” y que ambos son asociativos a izquierdas, ¿Cómo deben resolverse los conflictos en este estado?

**A** Ante un “and” se debe reducir y ante un “+” desplazar

- B** Ante un “and” se debe desplazar y ante un “+” reducir  
**C** Ante un “and” se debe desplazar y ante un “+” desplazar  
**D** Ante un “and” se debe reducir y ante un “+” reducir

La solución correcta es la **A**

2. (1,5 ptos.) Dada la siguiente gramática que genera árboles binarios de números enteros (en forma secuencial):

$$S \rightarrow ( \text{cte } S \ S ) \qquad S \rightarrow ( \text{cte} )$$

Diseñad un ETDS, asociado a esta gramática, para que calcule el número entero de mayor valor y la profundidad máxima del árbol de derivación. Por ejemplo, para la cadena: (2 (1 (0) (1)) (3 (2) (3))) el valor máximo sería el 3 y su profundidad 3.

$S \Rightarrow ( \text{cte } S \ S )$	$S.\text{max} = \max (\text{cte.val}, S^1.\text{max}, S^2.\text{max});$ $S.\text{prf} = \max (S^1.\text{prf}, S^2.\text{prf}) + 1;$
$S \Rightarrow ( \text{cte} )$	$S.\text{max} = \text{cte.val}; \quad S.\text{prf} = 1;$

3. (0,5 ptos.) Escribe las acciones semánticas para la comprobación de tipos de la siguiente expresión que accede a la posición “cte” del vector “id”:  $E \rightarrow \text{id} [\text{cte}]$

$E \Rightarrow \text{id} [\text{cte}]$	$\underline{\text{SI}} \neg [ \text{obtTds}(\text{id.n}, \text{id.t}) \wedge (\text{id.t} = \text{tvector}(\text{id.nel}, \text{id.tel})) \wedge$ $(\text{cte.t} = \text{tentero}) ] \{ E.t = \text{terror}; \text{MenError}(.); \}$ $\underline{\text{SINO}} \quad E.t = \text{id.tel};$
--	---

1<sup>er</sup> PARCIAL LENGUAJES DE PROGRAMACIÓN Y PROCESADORES DE LENGUAJES.  
2020-21

PARTE I: ANÁLISIS SINTÁCTICO DESCENDENTE.

SEGUNDA CONVOCATORIA

1. (1 pto.) Dada la siguiente gramática:

- 1:  $S \rightarrow x B S y$
- 2:  $S \rightarrow \epsilon$
- 3:  $A \rightarrow c A$
- 4:  $A \rightarrow \epsilon$
- 5:  $B \rightarrow b A$

f) (0.75 pto) Escribe en el cuadro de texto adjunto los conjuntos de SIG(S), SIG(A) y SIG(B).

SIG(S)={ \$, d}  
 SIG(A)={ a, d }  
 SIG(B)={ a, d }

g) (1 pto.) Construye la tabla de análisis LL(1). En el cuadro de respuestas indica las acciones, distintas de error, correspondientes a todos los no-terminales.

	a	b	c	d	\$
S	(a A S d, 1)			(ε, 2)	(ε, 2)
A		(b B, 3)			
B	(ε, 5)		(c B, 4)	(ε, 5)	

h) (1 pto.) A partir de la tabla de análisis LL(1) calculada anteriormente, realiza paso a paso la traza de análisis LL(1) para la cadena: **a b c d**

(S\$, abccd\$)      |- (aASd\$, abcd\$, 1)      |- (ASd\$, bccd\$, 1)      |- (bBSd\$, bccd\$, 1-3)  
 |- (BSd\$, ccd\$, 1-3)      |- (cBSd\$, ccd\$, 1-3-4)      |- (BSd\$, cd\$, 1-3-4)      |- (cBSd\$, cd\$, 1-3-4-4)  
 |- (BSd\$, d\$, 1-3-4-4)      |- (Sd\$, d\$, 1-3-4-4-5)      |- (d\$, d\$, 1-3-4-4-5-2)      |- (\$, \$, 1-3-4-4-5-2)

i) (0.25 pto) A la vista de la tabla de análisis, ¿Es una gramática LL(1)? Justifica tu respuesta

Si que es LL(1) porque no hay ninguna celda con más de una acción (derivación).

j) (0.5 pto) Tras aplicar el esquema de eliminación de recursión a izquierdas a la gramática

$A \rightarrow A x A$   
 $A \rightarrow A z$   
 $A \rightarrow z x$   
 $A \rightarrow \epsilon$

la gramática resultante es:

a) $A \rightarrow z x A'$ $A \rightarrow A'$ $A' \rightarrow x A A'$ $A' \rightarrow z A'$ $A' \rightarrow \varepsilon$	b) $A \rightarrow x A A'$ $A \rightarrow z A'$ $A' \rightarrow z x$ $A' \rightarrow \varepsilon$	c) $A \rightarrow x A A'$ $A \rightarrow z A'$ $A' \rightarrow z x$	d) $A \rightarrow z x A'$ $A \rightarrow A'$ $A' \rightarrow x A A'$ $A' \rightarrow z A'$
--	--	--	--

La solución correcta es la A