



**EXAMEN DE COMPILADORES (2º Grado en Informática, final febrero-2013)**

Apellidos, nombre:

DNI:

**Instrucciones:** Este enunciado y todos los folios usados deben entregarse al salir

**Parte I: PREGUNTAS TIPO TEST. 30%.** Cada dos respuestas incorrectas anulan una correcta.

1. Gracias a la teoría de *lenguajes formales*:

- a) El análisis léxico y sintáctico pueden realizarse de forma sistemática, puesto que consisten en la simulación de un autómata de pila y un autómata linealmente acotado, respectivamente.
- b) El análisis semántico puede realizarse de forma sistemática, pues todos los lenguajes de programación son sensibles al contexto y, por tanto, dicho análisis consiste exclusivamente en la simulación de un autómata finito.
- c) La primeras fases del proceso de análisis de un compilador se pueden realizar de forma eficiente y sistemática, pues suelen consistir en la simulación de autómatas deterministas.

2. El lenguaje de implementación de un compilador:

- a) Debe ser de bajo nivel, para que pueda funcionar en una máquina dada.
- b) En ningún caso puede ser el propio lenguaje fuente del compilador, puesto que no podríamos compilarlo de ninguna forma.
- c) Suele ser un lenguaje de alto nivel.

3. A partir de una expresión regular:

- a) No puede obtenerse directamente un AFD, sino que hay que convertirla primero en AFND y posteriormente convertir este AFND al AFD equivalente mediante la *construcción de subconjuntos*.
- b) Podemos obtener el AFND, aunque éste no puede simularse directamente, sino que habría que convertirlo previamente a AFD.
- c) Podemos obtener directamente el AFD o el AFND y simular cualquiera de los dos.

4. Considérese la siguiente gramática, recursiva por la izquierda:

$S \rightarrow A\alpha \mid \delta$

$A \rightarrow S\beta$

¿Cuál de las siguientes gramáticas es el resultado de eliminar correctamente la recursividad?

- a)  $S \rightarrow A\alpha \mid \delta$   
 $A \rightarrow \delta\beta \mid A\alpha\beta$
- b)  $S \rightarrow A\alpha \mid \delta$   
 $A \rightarrow \delta\beta \mid \delta\beta A'$   
 $A' \rightarrow \alpha\beta \mid \alpha\beta A'$
- c)  $S \rightarrow \alpha A \mid \delta$   
 $A \rightarrow \delta\beta \mid \delta\beta A'$   
 $A' \rightarrow \alpha\beta \mid \lambda$

5. Dada una especificación léxica de Flex con la siguiente lista de expresiones regulares:

$(00)^*$

$01^+$

$10^+$

¿Cuál de las siguientes cadenas no se podría procesar satisfactoriamente?

- a) 0111110
- b) 0001101
- c) 01100100

6. Dada la gramática con producciones:

$$E \rightarrow E + E \mid T$$

$$T \rightarrow T * F \mid F$$

$$F \rightarrow id \mid (E)$$

¿qué tipo de ambigüedades presenta?

- Sólo ambigüedad debida a la precedencia de los operadores + y \*.
- Sólo ambigüedad debida a la asociatividad del operador +.
- Ningún tipo de ambigüedad.

7. Dada la siguiente gramática:

$$\begin{array}{lcl} E & \rightarrow & id E' \\ E' & \rightarrow & \lambda \\ & & \wedge E' \\ & & . id E' \\ & & [ E ] E' \end{array}$$

La forma sentencial  $id^{\wedge}.id^{\wedge}E'$ :

- Es una forma sentencial derecha.
- Es una forma sentencial izquierda.
- Es una forma sentencial derecha e izquierda.

8. Una gramática LL:

- Tiene que ser propia, puesto que si no, sería imposible eliminar la recursividad por la izquierda.
- Puede no ser propia.
- Tiene que ser no recursiva por la izquierda, factorizada y  $\lambda$ -libre.

9. Una gramática LR:

- Tiene que ser propia.
- Puede no ser propia.
- Tiene que ser no ambigua y  $\lambda$ -libre.

10. Una tabla LALR:

- Tiene un tamaño intermedio entre una SLR y una LR-Canónica para la misma gramática.
- Tiene necesariamente que tener un tamaño menor que una LR-Canónica para la misma gramática.
- Tiene necesariamente el mismo tamaño que una tabla SLR para la misma gramática.

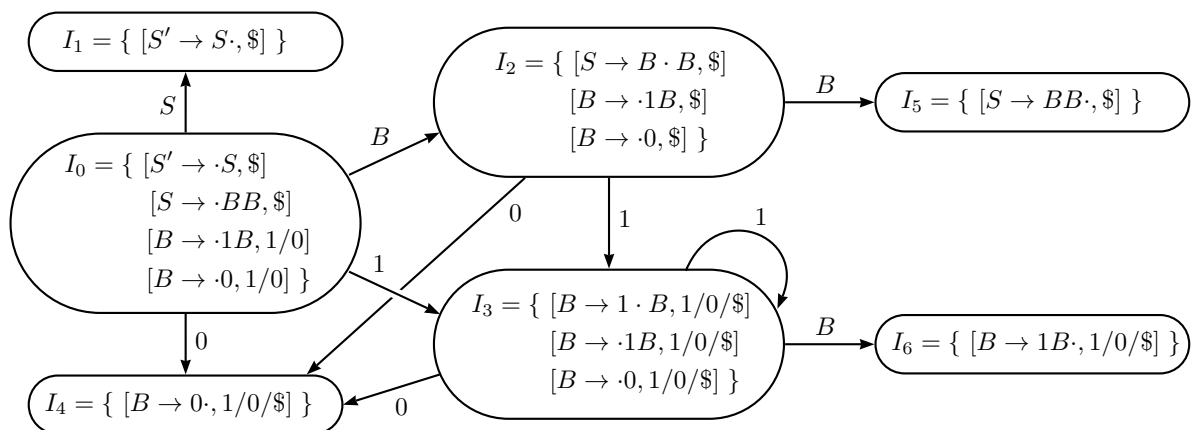
11. Supongamos que construimos un analizador LALR de la siguiente gramática:

$$S \rightarrow BB$$

$$B \rightarrow 1B$$

$$B \rightarrow 0$$

y obtenemos los siguientes conjuntos de ítems:



Indicar la respuesta correcta:

- a) Sólo se puede afirmar que la gramática es LALR.  
 b) Sólo se puede afirmar que la gramática es LR(1).  
 c) Se puede afirmar que la gramática es SLR y LR(1).
12. Continuando con el analizador LALR del ejemplo anterior, ¿en qué estado se encontraría el analizador al terminar de procesar la subcadena de entrada 011?
- a)  $I_3$   
 b)  $I_5$   
 c)  $I_6$
13. Los métodos generales de análisis sintáctico que analizan cualquier gramática libre de contexto:
- a) No existen.  
 b) No siempre pueden usarse puesto que requieren transformar las gramáticas previamente. Por ejemplo, el método CYK analiza gramáticas en *forma normal de Chomsky*.  
 c) No suelen usarse por falta de eficiencia.
14. Un *esquema de traducción*:
- a) Al igual que una *definición dirigida por la sintaxis*, no indica el orden en el que deben ejecutarse las acciones semánticas.  
 b) Ejecuta una acción semántica cuando se han procesado todos los símbolos que tiene a su izquierda.  
 c) No permite utilizar atributos ni introducir acciones semánticas en mitad de las reglas.
15. Dada la siguiente definición dirigida por la sintaxis:

PRODUCCIÓN	REGLAS SEMÁNTICAS
$S \rightarrow BC$	$C.base = B.base;$ $S.valor = C.valor;$
$C \rightarrow C_10$	$C_1.base = C.base;$ $C.valor = C_1.valor \times C_1.base;$
$C \rightarrow C_11$	$C_1.base = C.base;$ $C.valor = C_1.valor \times C_1.base + 1;$
$C \rightarrow 0$	$C.valor = 0;$
$C \rightarrow 1$	$C.valor = 1;$
$B \rightarrow b$	$B.base = 2;$
$B \rightarrow x$	$B.base = 16;$

Indicar la respuesta correcta:

- a) Se puede evaluar mediante un analizador LL(1), en una única pasada, sin hacer transformaciones a la gramática.  
 b) Se puede evaluar mediante un analizador LL(1) en una única pasada, haciendo transformaciones a la gramática.  
 c) No se puede evaluar mediante un analizador LR, porque tiene atributos heredados.

## Parte II: PREGUNTAS CORTAS. 15 %.

Dada la siguiente tabla LL(1):

	id	^	.	[	]	\$
E	$E \rightarrow id E'$					
E'		$E' \rightarrow ^ E'$	$E' \rightarrow . id E'$	$E' \rightarrow [ E ] E'$	$E' \rightarrow \lambda$	$E' \rightarrow \lambda$

asociada a la gramática:

$$\begin{array}{lcl}
 E & \rightarrow & id E' \\
 E' & \rightarrow & \lambda \\
 & | & ^ E' \\
 & | & . id E' \\
 & | & [ E ] E'
 \end{array}$$

- (0.25 puntos) Explicar por qué aparece la regla  $E' \rightarrow \lambda$  en las casillas  $\{E', \}$  y  $\{E', \$\}$
- (0.5 puntos) Rellenar la casilla  $\{E, \}$  con una llamada a una rutina de recuperación de errores *a nivel de frase*.
- (0.75 puntos) Proceder al reconocimiento o rechazo de la entrada  $id^{\wedge}.id[ ]\$$  usando recuperación de errores a nivel de frase si fuese necesario. En caso de no haber completado el apartado anterior, resolver el apartado empleando recuperación de errores en modo pánico.

## Parte III: PROBLEMA. 55 %.

Considerar la siguiente gramática:

$$\begin{array}{lcl}
 S & \rightarrow & LP ; E \\
 LP & \rightarrow & LP ; num \\
 & | & num \\
 E & \rightarrow & num \\
 & | & var \\
 & | & fun ( E )
 \end{array}$$

que genera frases consistentes en una lista de puntos (LP) y una expresión matemática sencilla en la que se emplean únicamente funciones **seno** y **coseno**. Por ejemplo, la gramática podría generar la siguiente cadena:

$$0,1;0,2;0,3;seno(seno(x))$$

- (0.5 puntos) Decir si se trata de una gramática LL(1), justificando la respuesta.
- (1 punto) Calcular los conjuntos PRIMERO y SIGUIENTE.
- (1.75 puntos) Calcular la tabla SLR(1) y comprobar si se trata de una gramática de este tipo.
- (0.5 puntos) ¿Es la gramática LALR? ¿Y LR-Canónica?
- (1.75 puntos) Supongamos que queremos construir un traductor simple para calcular derivadas de funciones seno y coseno, y evaluarlas en puntos concretos. Realizar una definición dirigida por la sintaxis para traducir una entrada generada por la gramática anterior en una lista de valores de la derivada de la expresión evaluada en los puntos que se han dado como entrada, junto con el texto de la derivada de la expresión. Cada valor debe separarse del anterior por punto y coma, e igualmente el texto de la derivada de la expresión. Recuerdese que las derivadas de **seno** y **coseno** son:

$$\begin{array}{ll}
 f(x) = \sin u & f'(x) = u' \cos u \\
 f(x) = \cos u & f'(x) = -u' \sin u
 \end{array}$$

Indicar el tipo de atributos semánticos utilizados, y describir las funciones auxiliares empleadas. Decorar el árbol sintáctico correspondiente a la entrada:  $0,1;0,2;0,3;seno(seno(x))$ .