LOGICA I LLENGUATGES

Curso 2020-2021

Examen final de teoría

(a) Explicar las diferentes maneras en que se puede definir el concepto de lenguaje regular.

(b) Explicar la relación existente entre los autómatas con pila y las gramáticas incontextuales.

(c) Definir los conceptos de árbol de derivación y gramática ambigua, y explicar por qué las gramáticas ambiguas no pueden utilizarse en el diseño de compiladores.

(d) Explicar cómo se construye la tabla de análisis de una gramática incontextual, y definir el concepto de gramática LL(1).

(e) Explicar en qué consisten las tres fases del diseño de un compilador, y en qué fases se utilizan los autómatas deterministas, los autómatas con pila y las gramáticas incontextuales.

Determinar entonces, razonando la respuesta, si las siguientes fórmulas son verdaderas o falsas en I:

$\varphi_1 = \forall x Pxx$,

$\varphi_2 = \exists y \neg Qay$,

$\varphi_3 = \exists x(Pf(x) \wedge Qxf(a))$,

$\varphi_4 = \forall x \exists y(Py \wedge Qxy)$,

$\varphi_5 = \forall y \exists x Qf(x)y \to \exists x \forall y Qf(x)y$.

(7,5 puntos)

(b) Consideremos el vocabulario $\sigma = \{P^1, Q^1\}$. Demostrar que las fórmulas $\psi_1 = \forall x(Px \vee Qx)$ y $\psi_2 = \forall x Px \vee \forall x Qx$ no son lógicamente equivalentes.

(2,5 puntos)

Problema 3. Consideremos el autómata indeterminista $M = (\{A, B, C, D, E, F\}, \{0, 1\}, \Delta, A, \{D, F\})$ donde Δ está definida por la siguiente tabla:

A	0	A
A	1	A
A	1	C
A	λ	B
B	0	E
B	1	C
C	0	D
D	λ	F
E	1	F
F	0	D

Se pide entonces:

(1) Siguiendo el método visto en clase, trasformar el autómata M en un autómata determinista equivalente. (7 puntos)

(2) Programar en Java el autómata determinista obtenido en (1).

(3 puntos)

Problema 4. La siguiente gramática incontextual G genera una clase de instrucciones en un lenguaje de programación.

1. $S \longrightarrow \underline{do}\, Y\ \underline{while}\,(C);$

2. $Y \longrightarrow \underline{id}\ =\ E;$

3. $E \longrightarrow E * F$

4. $E \longrightarrow E/F$

5. $E \longrightarrow F$

6. $F \longrightarrow (E)$

7. $F \longrightarrow \underline{id}$

8. $F \longrightarrow \underline{int}$

9. $F \longrightarrow \underline{float}$

10. $C \longrightarrow C\ \&\&\ D$

11. $C \longrightarrow D$

12. $D \longrightarrow \underline{id}\ >=\ \underline{id}$

13. $D \longrightarrow \underline{id}\ >\ \underline{id}$

Se pide entonces:

(a) Dar una derivación en G para la palabra $\underline{do}\ \ \underline{id}\ =\ \underline{id} + \underline{float} - \underline{int}\ \ \underline{while}\ (\underline{id} > \underline{id}\ \&\&\ \underline{id} >= \underline{id});$

(1,5 puntos)

(b) Siguiendo el método visto en clase, construir el autómata con pila M asociado a G. (2,5 puntos)

(c) Explicar por qué G no es una gramática LL(1).

(1 punto)

(d) Aplicar las reglas de factorización y recursión a la gramática G.

(2 puntos)

(e) Construir la tabla de análisis de la gramática obtenida en (d).

(3 puntos)

Problema 1. Consideremos las siguientes gramáticas incontextuales G_1 y G_2. La gramática G_1 está definida por las siguientes producciones:

1. $S \longrightarrow 0S1S$.
2. $S \longrightarrow 1S0S$.
3. $S \longrightarrow \lambda$.

Y la gramática G_2 está definida por las producciones siguientes:

1. $S \longrightarrow 0$.
2. $S \longrightarrow S0$.
3. $S \longrightarrow 1SS$.
4. $S \longrightarrow SS1$.
5. $S \longrightarrow S1S$.

Se pide entonces:

(a) Dar una derivación en G_1 que genere la palabra 1010 y una derivación en G_2 que genere la palabra 10010.

(b) Determinar si G_1,G_2 son ambiguas, razonando la respuesta.

(c) Describir los lenguajes $L(G_1)$ y $L(G_2)$.

(d) Aplicando el método visto en clase, construir el autómata con pila equivalente a G_2.

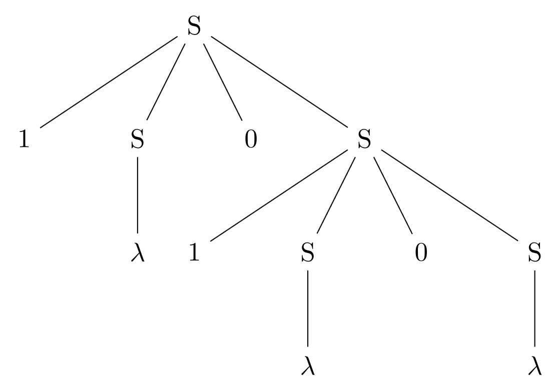
(e) Dar un cómputo en el autómata construido en (d) que reconozca la palabra 10010.

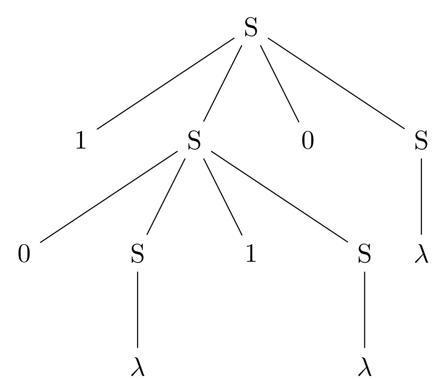
Solución:

(a) $S \Rightarrow^2 1S0S \Rightarrow^1 10S1S0S \Rightarrow^3 10S1S0 \Rightarrow^3 10S10 \Rightarrow^3 1010$

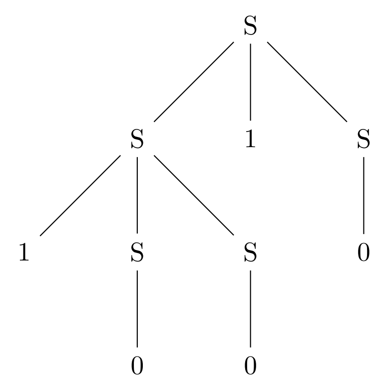
$S \Rightarrow^3 1SS \Rightarrow^4 1SS1S \Rightarrow^1 10S1S \Rightarrow^1 1001S \Rightarrow^1 10010$

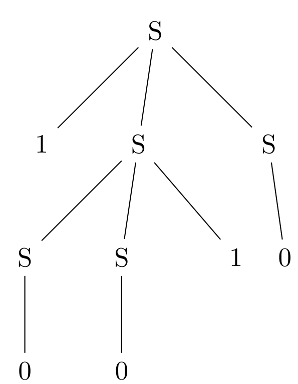
(b) Las gramáticas G_1 y G_2 son ambiguas, porque cada una de las dos palabras consideradas en el apartado (a) tiene dos áboles de derivación. La palabra 1010 tiene los dos siguientes árboles de derivación en G_1:





Y la palabra 10010 tiene los dos siguientes árboles de derivación en G_2:





(c) Tenemos que $L(G_1) = \{x \in \{0,1\}^* : n_0(x) = n_1(x)\}$ y $L(G_2) = \{x \in \{0,1\}^* : n_0(x) > n_1(x)\}$.

(d) $M = (K, \Sigma, \Gamma, \Delta, q_0, F)$, donde el conjunto de los estados es $K = \{q_0, f\}$, el vocabulario de la cinta es $\Sigma = \{0, 1\}$, el vocabulario de la pila es $\Gamma = \{0, 1, S\}$, el estado inicial es q_0, el único estado aceptador es f y Δ está formado por las siguientes transiciones:

1. $((q_0, \lambda, \lambda), (f, S))$.
2. $((f, \lambda, S), (f, 0))$.
3. $((f, \lambda, S), (f, S0))$.
4. $((f, \lambda, S), (f, 1SS))$.
5. $((f, \lambda, S), (f, SS1))$.
6. $((f, \lambda, S), (f, S1S))$.
7. $((f, 0, 0), (f, \lambda))$.
8. $((f, 1, 1), (f, \lambda))$.

(e) Cómputo de M que reconoce la palabra 10010:

estado	cinta	pila	transición
q_0	10010	λ	–
f	10010	S	1
f	10010	$1SS$	4
f	0010	SS	8
f	0010	$0S$	2
f	010	S	7
f	010	$S1S$	6
f	010	$01S$	2
f	10	$1S$	7
f	0	S	8
f	0	0	2
f	λ	λ	7

Problema 2. Consideremos la siguiente gramática incontextual G para diseñar una calculadora de dígitos decimales, donde E es el símbolo inicial.

1. $E \longrightarrow T$
2. $E \longrightarrow EOE$
3. $T \longrightarrow A$
4. $T \longrightarrow TPA$
5. $O \longrightarrow +$
6. $O \longrightarrow -$
7. $P \longrightarrow *$
8. $P \longrightarrow /$
9. $A \longrightarrow \underline{int}$
10. $A \longrightarrow \underline{float}$

Se pide entonces:

(a) Demostrar que G es ambigua.

(b) Escribir una gramática equivalente a G que no sea ambigua.

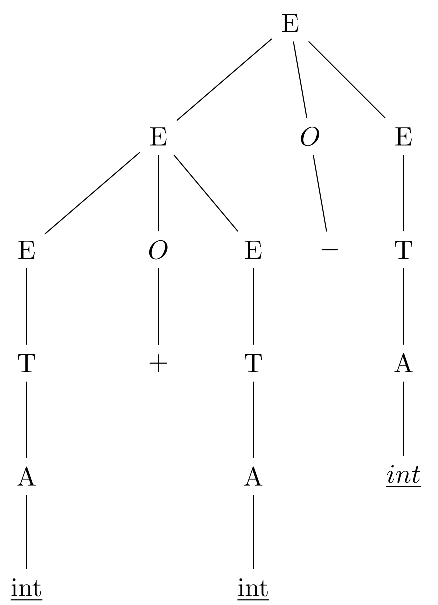
(c) Aplicando el método visto en clase, construir el autómata con pila equivalente a la gramática del apartado (b).

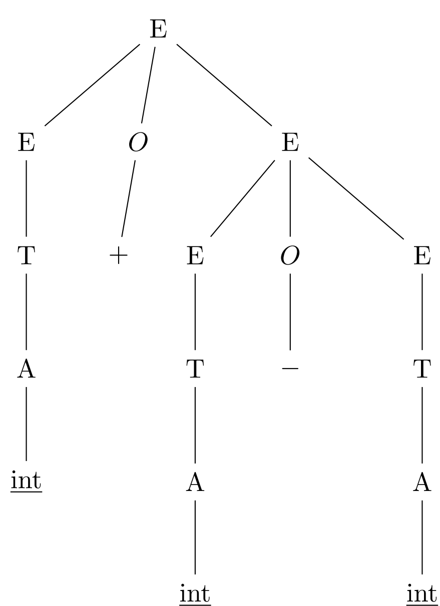
(d) Aplicando el método visto en clase, transformar la gramática del apartado (b) en una gramática equivalente LL(1).

(e) Construir la tabla de análisis de la gramática obtenida en (d).

Solución:

(a) Para demostrar que G es ambigua, consideremos la palabra $x = \underline{int} + \underline{int} - \underline{int}$. La palabra x tiene entonces los dos siguientes árboles de derivación:


E
E
O
E
E
O
E
−
T
T
+
T
A
A
A
int
int
int


E
E
O
E
T
+
E
O
E
A
T
−
T
int
A
A
int
int

(b) La gramática G es ambigua, porque en la parte derecha de la producción 2 de G aparece la variable E repetida. Para eliminar entonces la ambigüedad de G reemplazamos una de las dos apariciones de la variable E en la parte derecha de la producción 2 por la variable T (evitando de esta forma la repetición de variables). Obtenemos la siguiente gramática G':

1. $E \longrightarrow T$
2. $E \longrightarrow TOE$
3. $T \longrightarrow A$
4. $T \longrightarrow TPA$
5. $O \longrightarrow +$
6. $O \longrightarrow -$
7. $P \longrightarrow *$
8. $P \longrightarrow /$
9. $A \longrightarrow \underline{int}$
10. $A \longrightarrow \underline{float}$

En G', la variable E genera de manera unívoca una suma/resta de términos y la variable T genera también de manera unívoca un producto/división de factores, que pueden ser o bien números enteros (tipo $\underline{int}$) o bien números decimales (tipo $\underline{float}$). Por tanto, G' no es ambigua.

(c) $M = (K, \Sigma, \Gamma, \Delta, q_0, F)$, donde el conjunto de los estados es $K = \{q_0, f\}$, el vocabulario de la cinta es $\Sigma = \{+, -, *, /, \underline{int}, \underline{float}\}$, el vocabulario de la pila es $\Gamma = \{+, -, *, /, \underline{int}, \underline{float}, E, T, A, O, P\}$, el estado inicial es q_0, el único estado aceptador es f y Δ está formado por las siguientes transiciones:

1. $((q_0, \lambda, \lambda), (f, S))$.
2. $((f, \lambda, E), (f, T))$.
3. $((f, \lambda, E), (f, TOE))$.
4. $((f, \lambda, T), (f, A))$.
5. $((f, \lambda, T), (f, TPA))$.
6. $((f, \lambda, O), (f, +))$.
7. $((f, \lambda, O), (f, -))$.

8. $((f, \lambda, P), (f, *))$.
9. $((f, \lambda, P), (f, /))$.
10. $((f, \lambda, A), (f, \underline{int}))$.
11. $((f, \lambda, A), (f, \underline{float}))$.
12. $((f, +, +), (f, \lambda))$.
13. $((f, -, -), (f, \lambda))$.
14. $((f, *, *), (f, \lambda))$.
15. $((f, /, /), (f, \lambda))$.
16. $((f, \underline{int}, \underline{int}), (f, \lambda))$.
17. $((f, \underline{float}, \underline{float}), (f, \lambda))$.

(d) Aplicando la regla de factorización, reemplazamos las producciones $E \longrightarrow T$ y $E \longrightarrow TOE$ por las producciones $E \longrightarrow TX$, $X \longrightarrow \lambda$ y $X \longrightarrow OE$. Y aplicando la la regla de recursión, reemplazamos las producciones $T \longrightarrow A$ y $T \longrightarrow TPA$ por las producciones $T \longrightarrow AY$, $Y \longrightarrow PAY$ e $Y \longrightarrow \lambda$.

Por tanto, obtenemos la siguiente gramática $G^{''}$ equivalente a G':

1. $E \longrightarrow TX$
2. $X \longrightarrow \lambda$
3. $X \longrightarrow OE$
4. $T \longrightarrow AY$
5. $Y \longrightarrow PAY$
6. $Y \longrightarrow \lambda$
7. $O \longrightarrow +$
8. $O \longrightarrow -$
9. $P \longrightarrow *$
10. $P \longrightarrow /$
11. $A \longrightarrow \underline{int}$
12. $A \longrightarrow \underline{float}$

(e) La tabla de análisis de G'' es la siguiente:

TABLA	$+$	$-$	$*$	$/$	<u>int</u>	<u>float</u>
E					1	1
X	3	3				
T					4	4
Y	6	6	5	5		
O	7	8				
P			9	10		
A					11	12

Obsérvese que Siguientes$(X) = \emptyset$. Por tanto, la producción 2 no aparece en la tabla de análisis.

Y de las derivaciones

$E \Rightarrow^1 TX \Rightarrow^3 TOE \Rightarrow^4 AYOE \Rightarrow^7 AY + E,$

$E \Rightarrow^1 TX \Rightarrow^3 TOE \Rightarrow^4 AYOE \Rightarrow^8 AY - E$

se deduce que $+, - \in$ Siguientes(Y) y, por tanto, la producción 6 pertenece a TABLA$(Y, +)$ y a TABLA$(Y, -)$.

Problema 3. La siguiente gramática incontextual G genera una clase de declaraciones de JAVA.

1. $S \longrightarrow ES$
2. $S \longrightarrow E$
3. $E \longrightarrow TF;$
4. $T \longrightarrow \underline{int}$
5. $T \longrightarrow \underline{int}\,[\,]$
6. $T \longrightarrow \underline{float}$
7. $T \longrightarrow \underline{float}\,[\,]$
8. $F \longrightarrow F\,,\,\underline{id}$
9. $F \longrightarrow \underline{id}$

Se pide entonces:

(a) Dar una derivación en G para la palabra

$\underline{int}\ \ \underline{id}\ ,\ \underline{id}\ ;\ \underline{float}\,[\,]\ \ \underline{id}\ ;$

(b) Siguiendo el método visto en clase, construir el autómata con pila M asociado a G.

(c) Dar un cómputo en M que reconozca la palabra

$\underline{int}\,\underline{id}\,;\underline{float}\,[\,]\ \ \underline{id};$

(d) Explicar por qué G no es una gramática LL(1).

(e) Aplicar las reglas de factorización y recursión para transformar la gramática G en una gramática LL(1).

(f) Construir la tabla de análisis de la gramática obtenida en (e).

Solución:

(a) $S \Rightarrow^1 ES \Rightarrow^2 EE \Rightarrow^3 TF;E \Rightarrow^3 TF;TF; \Rightarrow^4 \underline{int}\,F;TF; \Rightarrow^8 \underline{int}\,F,\underline{id};TF;$ $\Rightarrow^9 \underline{int}\,\underline{id},\underline{id};TF; \Rightarrow^7 \underline{int}\,\underline{id},\underline{id};\underline{float}\,[\,]F; \Rightarrow^9 \underline{int}\,\underline{id},\underline{id};\ \underline{float}\,[\,]\,\underline{id};$

(b) $M = (K, \Sigma, \Gamma, \Delta, q_0, F)$, donde el conjunto de los estados es $K = \{q_0, f\}$, el vocabulario de la cinta es $\Sigma = \{\underline{id}, \underline{int}, \underline{float}, ;, ,, [,]\}$, el vocabulario de la pila es $\Gamma = \Sigma \cup V$ siendo $V = \{S, E, T, F\}$, el estado inicial es q_0, el único estado aceptador es f y Δ está formado por las siguientes transiciones:

1. $((q_0, \lambda, \lambda), (f, S))$.

2. $((f,\lambda,S),(f,ES))$.
3. $((f,\lambda,S),(f,E))$.
4. $((f,\lambda,E),(f,TF;))$.
5. $((f,\lambda,T),(f,\underline{int}))$.
6. $((f,\lambda,T),(f,\underline{int}[\,]))$.
7. $((f,\lambda,T),(f,\underline{float}))$.
8. $((f,\lambda,T),(f,\underline{float}[\,]))$.
9. $((f,\lambda,F),(f,F\ ,\ \underline{id}))$.
10. $((f,\lambda,F),(f,\underline{id}))$.
11. $((f,\underline{id},\underline{id}),(f,\lambda))$.
12. $((f,\underline{int},\underline{int}),(f,\lambda))$.
13. $((f,\underline{float},\underline{float}),(f,\lambda))$.
14. $((f,;,;),(f,\lambda))$.
15. $((f,,,,),(f,\lambda))$.
16. $((f,[,[),(f,\lambda))$.
17. $((f,],]),(f,\lambda))$.

(c) Cómputo que reconoce $\underline{int}\ \underline{id}\,;\underline{float}\,[\,]\ \ \underline{id};$

estado	cinta	pila	transición
q_0	$\underline{int}\,\underline{id}\,;\underline{float}\,[\,]\ \underline{id}\,;$	λ	–
f	$\underline{int}\,\underline{id}\,;\underline{float}\,[\,]\ \underline{id}\,;$	S	1
f	$\underline{int}\,\underline{id}\,;\underline{float}\,[\,]\ \underline{id}\,;$	ES	2
f	$\underline{int}\,\underline{id}\,;\underline{float}\,[\,]\ \underline{id}\,;$	$TF;S$	4
f	$\underline{int}\,\underline{id}\,;\underline{float}\,[\,]\ \underline{id}\,;$	$\underline{int}\,F;S$	5
f	$\underline{id};\underline{float}\,[\,]\ \underline{id}\,;$	$F;S$	12
f	$\underline{id};\underline{float}\,[\,]\ \underline{id}\,;$	$\underline{id};S$	10
f	$;\underline{float}\,[\,]\ \underline{id}\,;$	$;S$	11
f	$\underline{float}\,[\,]\ \underline{id}\,;$	S	14
f	$\underline{float}\,[\,]\ \underline{id};$	E	3
f	$\underline{float}\,[\,]\ \underline{id}\,;$	$TF;$	4
f	$\underline{float}\,[\,]\ \underline{id}\,;$	$\underline{float}\,[\,]F;$	8
f	$[\,]\ \underline{id}\,;$	$[\,]F;$	13
f	$]\ \underline{id}\,;$	$]F;$	16
f	$\underline{id}\,;$	$F;$	17
f	$\underline{id}\,;$	$\underline{id}\,;$	10
f	$;$	$;$	11
f	λ	λ	14

(d) La gramática G no es LL(1), porque hay conflictos al construir su tabla de análisis. Por ejemplo, las producciones $1,2 \in$ TABLA$(S,\underline{int})$, ya que $\underline{int} \in$ Primeros(E).

(e) Aplicando la regla de factorización, reemplazamos las producciones $S \longrightarrow ES$, $S \longrightarrow E$ por las producciones $S \longrightarrow ES'$, $S' \longrightarrow S$, $S' \longrightarrow \lambda$. Aplicando otra vez la regla de factorización, reemplazamos las producciones $T \longrightarrow \underline{int}$, $T \longrightarrow \underline{int}\,[\,]$ por las producciones $T \longrightarrow \underline{int}\,T'$, $T' \longrightarrow \lambda$, $T' \longrightarrow [\,]$. Y aplicando de nuevo la regla de factorización, reemplazamos las producciones $T \longrightarrow \underline{float}$, $T \longrightarrow \underline{float}\,[\,]$ por las producciones $T \longrightarrow \underline{float}\,T''$, $T'' \longrightarrow \lambda$, $T'' \longrightarrow [\,]$. Por último, aplicando la regla de recursión, reemplazamos las producciones $F \longrightarrow F,\underline{id}$, $F \longrightarrow \underline{id}$ por las producciones $F \longrightarrow \underline{id}\,F'$, $F' \longrightarrow ,\underline{id}\,F'$, $F' \longrightarrow \lambda$.

Se observa que las variables T' y T'' son equivalentes, ya que generan el mismo lenguaje, el formado por las palabras λ y $[\,]$. Por tanto, podemos identificar las dos variables, y utilizar únicamente una de ellas, por ejemplo la variable T'. Obtenemos entonces la siguiente gramática G' equivalente a G:

1. $S \longrightarrow ES'$
2. $S' \longrightarrow S$
3. $S' \longrightarrow \lambda$

4. $E \longrightarrow TF;$
5. $T \longrightarrow \underline{int}\, T'$
6. $T' \longrightarrow \lambda$
7. $T' \longrightarrow [\,]$
8. $T \longrightarrow \underline{float}\, T'$
9. $F \longrightarrow \underline{id}\, F'$
10. $F' \longrightarrow , \underline{id}\, F'$
11. $F' \longrightarrow \lambda$

(f) La tabla de análisis de G' es la siguiente:

TABLA	id	int	float	;	,	[	]
S		1	1				
S'		2	2				
E		4	4				
T		5	8				
T'	6					7	
F	9						
F'				11	10		

Como Siguientes$(S') = \emptyset$, la producción 3 no aparece en la tabla de análisis.

Obsérvese que de la derivación

$S \Rightarrow^1 ES' \Rightarrow^4 TF; S' \Rightarrow^5 \underline{int}\, T' F; S' \Rightarrow^9 \underline{int}\, T' \underline{id}\, F'; S'$

se deduce que $\underline{id} \in$ Siguientes(T') y, por tanto, la producción 6 $\in$ TABLA$(T', \underline{id})$.

Y de la derivación

$S \Rightarrow^1 ES' \Rightarrow^4 TF; S' \Rightarrow^9 T \underline{id}\, F'; S'$

se deduce que $; \in$ Siguientes(F') y, por tanto, la producción 11 pertenece a TABLA$(F', ;)$.

Problema 4. La siguiente gramática incontextual G genera una clase de instrucciones repetitivas de JAVA.

1. $S \longrightarrow \underline{do}\, Y\, \underline{while}\,(C)$
2. $Y \longrightarrow \underline{id} \;=\; E;$
3. $E \longrightarrow E * F$
4. $E \longrightarrow E/F$
5. $E \longrightarrow F$
6. $F \longrightarrow (E)$
7. $F \longrightarrow \underline{id}$
8. $F \longrightarrow \underline{int}$
9. $F \longrightarrow \underline{float}$
10. $C \longrightarrow C \;\&\&\; D$
11. $C \longrightarrow D$
12. $D \longrightarrow \underline{id} >= \underline{id}$
13. $D \longrightarrow \underline{id} > \underline{id}$

Se pide entonces:

(a) Dar una derivación en G para la palabra $\underline{do}\, \underline{id} = \underline{int} * \underline{float}\, / \,\underline{id}\, ;\, \underline{while}\; (\underline{id} > \underline{id} \;\&\&\; \underline{id} >= \underline{id})$

(b) Siguiendo el método visto en clase, construir el autómata con pila M asociado a G.

(c) Explicar por qué G no es una gramática LL(1).

(d) Aplicar las reglas de factorización y recursión a la gramática G.

(e) Construir la tabla de análisis de la gramática obtenida en (d).

Solución:

(a) $S \Rightarrow^1 \underline{do}\, Y\, \underline{while}\,(C) \Rightarrow^{10} \underline{do}\, Y\, \underline{while}\,(C \;\&\&\; D) \Rightarrow^{11} \underline{do}\, Y\, \underline{while}\,(D \;\&\&\; D) \Rightarrow^{13} \underline{do}\, Y\, \underline{while}\,(\underline{id} > \underline{id} \;\&\&\; D) \Rightarrow^{12} \underline{do}\, Y\, \underline{while}\,(\underline{id} > \underline{id} \;\&\&\; \underline{id} >= \underline{id}) \Rightarrow^2 \underline{do}\, \underline{id} = E;\, \underline{while}\,(\underline{id} > \underline{id} \;\&\&\; \underline{id} >= \underline{id}) \Rightarrow^4 \underline{do}\, \underline{id} = E/F;\, \underline{while}\,(\underline{id} > \underline{id} \;\&\&\; \underline{id} >= \underline{id}) \Rightarrow^3 \underline{do}\, \underline{id} = E{*}F/F;\, \underline{while}\,(\underline{id} > \underline{id} \;\&\&\; \underline{id} >= \underline{id}) \Rightarrow^5 \underline{do}\, \underline{id} = F * F/F;\, \underline{while}\,(\underline{id} > \underline{id} \;\&\&\; \underline{id} >= \underline{id}) \Rightarrow^8 \underline{do}\, \underline{id} = \underline{int} * F/F;\, \underline{while}\,(\underline{id} > \underline{id} \;\&\&\; \underline{id} >= \underline{id}) \Rightarrow^9 \underline{do}\, \underline{id} = \underline{int} * \underline{float}\,/F;\, \underline{while}\,(\underline{id} > \underline{id} \;\&\&\; \underline{id} >= \underline{id}) \Rightarrow^7 \underline{do}\, \underline{id} = \underline{int} * \underline{float}\,/\,\underline{id}\,;\, \underline{while}\,(\underline{id} > \underline{id} \;\&\&\; \underline{id} >= \underline{id})$

(b) Tenemos que $M = (K, \Sigma, \Gamma, \Delta, q_0, F)$, donde el conjunto de los estados es $K = \{q_0, f\}$, el vocabulario de la cinta es $\Sigma = \{\underline{do}, \underline{while}, \underline{id}, \underline{int}, \underline{float}, =, ;, *, /, (,), >, >=, \&\&\}$, el vocabulario de la pila es $\Gamma = \Sigma \cup V$ siendo $V = \{S, Y, E, F, C, D\}$, el estado inicial es q_0, el único estado aceptador es f y Δ está formado por las siguientes transiciones:

1. $((q_0, \lambda, \lambda), (f, S))$.
2. $((f, \lambda, S), (f, \underline{do}\,\{Y\}\,\underline{while}\,(C)))$.
3. $((f, \lambda, Y), (f, \underline{id} = E;))$.
4. $((f, \lambda, E), (f, E * F))$.
5. $((f, \lambda, E), (f, E/F))$.
6. $((f, \lambda, E), (f, F))$.
7. $((f, \lambda, F), (f, (E)))$.
8. $((f, \lambda, F), (f, \underline{id}))$.
9. $((f, \lambda, F), (f, \underline{int}))$.
10. $((f, \lambda, F), (f, \underline{float}))$.
11. $((f, \lambda, C), (f, C\&\&D))$.
12. $((f, \lambda, C), (f, D))$.
13. $((f, \lambda, D), (f, \underline{id} >= \underline{id}))$.
14. $((f, \lambda, D), (f, \underline{id} > \underline{id}))$.
15. $((f, \underline{do}, \underline{do}), (f, \lambda))$.
16. $((f, \underline{while}, \underline{while}), (f, \lambda))$.
17. $((f, \underline{id}, \underline{id}), (f, \lambda))$.
18. $((f, \underline{int}, \underline{int}), (f, \lambda))$.
19. $((f, \underline{float}, \underline{float}), (f, \lambda))$.
20. $((f, =, =), (f, \lambda))$.
21. $((f, ;, ;), (f, \lambda))$.
22. $((f, *, *), (f, \lambda))$.
23. $((f, /, /), (f, \lambda))$.
24. $((f, \&\&, \&\&), (f, \lambda))$.

25. $((f, (, ()), (f, \lambda))$.

26. $((f,),)), (f, \lambda))$.

27. $((f, >=, >=), (f, \lambda))$.

28. $((f, >, >), (f, \lambda))$.

(c) La gramática G no es LL(1), porque hay conflictos al construir su tabla de análisis. Por ejemplo, las producciones $12, 13 \in$ TABLA$[D, \underline{id}]$, ya que $\underline{id}$ es el primer símbolo de las partes derechas de las producciones 12 y 13.

(d) Aplicando la regla de recursión, reemplazamos las producciones $E \longrightarrow E * F$, $E \longrightarrow E/F$ y $E \longrightarrow F$ por las producciones $E \longrightarrow FE'$, $E' \longrightarrow *FE'$, $E' \longrightarrow /FE'$ y $E' \longrightarrow \lambda$.

Aplicando de nuevo la regla de recursión, reemplazamos las producciones $C \longrightarrow C$ && D y $C \longrightarrow D$ por las producciones $C \longrightarrow DC'$, $C' \longrightarrow$ && DC' y $C' \longrightarrow \lambda$.

Por último, aplicando la regla de factorización, reemplazamos las producciones $D \longrightarrow \underline{id} >= \underline{id}$ y $D \longrightarrow \underline{id} > \underline{id}$ por las producciones $D \longrightarrow \underline{id}\, D'$, $D' \longrightarrow >= \underline{id}$ y $D' \longrightarrow > \underline{id}$.

Por tanto, obtenemos la siguiente gramática G' equivalente a G:

1. $S \longrightarrow \underline{do}\, Y\, \underline{while}\, (C)$
2. $Y \longrightarrow \underline{id} = E;$
3. $E \longrightarrow FE'$
4. $E' \longrightarrow *FE'$
5. $E' \longrightarrow /FE'$
6. $E' \longrightarrow \lambda$
7. $F \longrightarrow (E)$
8. $F \longrightarrow \underline{id}$
9. $F \longrightarrow \underline{int}$
10. $F \longrightarrow \underline{float}$
11. $C \longrightarrow DC'$
12. $C' \longrightarrow$ && DC'
13. $C' \longrightarrow \lambda$

14. $D \longrightarrow \underline{id}\, D'$

15. $D' \longrightarrow >= \ \underline{id}$

16. $D' \longrightarrow > \ \underline{id}$

(e) La tabla de análisis de G' es la siguiente:

TABLA	$\underline{\text{do}}$	$\underline{\text{while}}$	$\underline{\text{id}}$	$\underline{\text{int}}$	$\underline{float}$	(	)	=	;	$*$	/	>=	>	&&
S	1													
Y			2											
E			3	3	3	3								
E'							6		6	4	5			
F			8	9	10	7								
C			11											
C'							13							12
D			14											
D'												15	16	

Obsérvese que $1 \in$ TABLA$[S, \underline{do}]$, porque $\underline{do}$ es el primer símbolo de la parte derecha de la regla 1. Por el mismo motivo, tenemos que $2 \in$ TABLA$[Y, \underline{id}]$, $4 \in$ TABLA$[E', *]$, $5 \in$ TABLA$[E', /]$, $7 \in$ TABLA$[F,)]$, $8 \in$ TABLA$[F, \underline{id}]$, $9 \in$ TABLA$[F, \underline{int}]$, $10 \in$ TABLA$[F, \underline{float}]$, $12 \in$ TABLA$[C', \&\&]$, $14 \in$ TABLA$[D, \underline{id}]$, $15 \in$ TABLA$[D', >=]$ y $16 \in$ TABLA$[D', >]$.

Tenemos que 3 pertenece a TABLA$[E, \underline{id}]$, a TABLA$[E, \underline{int}]$, a TABLA$[E, \underline{float}]$ y a TABLA$[E,)]$, porque Primeros(E) = Primeros(F) = $\{(, \underline{id}, \underline{int}, \underline{float}\}$. Y tenemos que $11 \in$ TABLA$[C, \underline{id}]$, porque Primeros(C) = Primeros(D) = $\{\underline{id}\}$.

Por otra parte, de la derivación

$S \Rightarrow^1 \underline{do}\, Y\, \underline{while}\, (C) \Rightarrow^2 \underline{do}\, \underline{id} = E;\, \underline{while}\, (C) \Rightarrow^3 \underline{do}\, \underline{id} = FE';\, \underline{while}\, (C)$

se deduce que ; Siguientes(E') y, por tanto, tenemos que la producción $6 \in$ TABLA$[E', ;]$.

Y de la derivación

$S \Rightarrow^1 \underline{do}\, Y\, \underline{while}\, (C) \Rightarrow^2 \underline{do}\, \underline{id} = E;\, \underline{while}\, (C) \Rightarrow^3 \underline{do}\, \underline{id} = FE';\, \underline{while}\, (C) \Rightarrow^7 \underline{do}\, \underline{id} = (E)E';\, \underline{while}\, (C) \Rightarrow^3 \underline{do}\, \underline{id} = (FE')E';\, \underline{while}\, (C)$

se deduce que) Siguientes(E') y, por tanto, tenemos que la producción $6 \in$ TABLA$[E',)]$.

Por último, de la derivación

$S \Rightarrow^1 \underline{do}\, Y\, \underline{while}\, (C) \Rightarrow^{11} \underline{do}\, Y\, \underline{while}\, (DC')$

deducimos que) Siguientes(C') y, por tanto, tenemos que la producción 13 $\in$ TABLA$[C',)]$.

Como no hay conflictos en la tabla de análisis de G', tenemos que G' es LL(1).

Problema 5. Una combinación para enteros sin signo en el lenguaje LISP es ó bien un entero sin signo ó bien una expresión que representa la aplicación de un operador a una serie de argumentos numéricos, que tiene el siguiente formato:

$$(\underline{\text{Operador}}\ \ \underline{\text{Operando}}_1\ \ \ldots\ \ \underline{\text{Operando}}_n)$$

donde $n > 0$, los operadores son $+, -, *$ y $/$, y los operandos son combinaciones para enteros sin signo. Las siguientes expresiones son ejemplos de combinaciones:

(+ 1 2),
(+ 1 2 3),
(+ 1 2 (+ 1 2)),
(+ 1 (+ 1 1) (+ 1 2)),
$(+\ (-\ (*\ 4\ \ 10)(+\ (*\ 2\ \ 19)\ 1))(/\ 8\ 4)\ 3)$.

Representamos por $\underline{int}$ a la categoría sintáctica de los números enteros sin signo. La siguiente gramática incontextual G genera entonces el lenguaje de las combinaciones de LISP.

1. $C \longrightarrow (OA)$.
2. $C \longrightarrow \underline{int}$.
3. $O \longrightarrow +$.
4. $O \longrightarrow -$.
5. $O \longrightarrow *$.
6. $O \longrightarrow /$.
7. $A \longrightarrow AC$.
8. $A \longrightarrow C$.

donde C es la variable inicial.

Se pide entonces:

(a) Dar una derivación en G que genere la palabra $(+\ \underline{int}\ (-\ \underline{int}\ \underline{int})\ (*\ \underline{int}\ \underline{int}\ \underline{int}))$.

(b) Aplicando el método visto en clase, construir el autómata con pila M asociado a G.

(c) Dar un cómputo en el autómata construido en (b) que reconozca la palabra $(+\ \underline{int}\ (-\ \underline{int}\ \underline{int}))$.

(d) Aplicando el método visto en clase, transformar la gramática G en una gramática LL(1).

(e) Construir la tabla de análisis de la gramática obtenida en (d).

Solución:

(a) $C \Rightarrow^1 (OA) \Rightarrow^3 (+A) \Rightarrow^7 (+AC) \Rightarrow^7 (+ACC) \Rightarrow^8 (+CCC) \Rightarrow^2$ $(+\underline{int}\,CC) \Rightarrow^1 (+\underline{int}\,(OA)C) \Rightarrow^4 (+\underline{int}(-A)C) \Rightarrow^{7,8} (+\underline{int}\,(-CC)C) \Rightarrow^2 (+\underline{int}(-\underline{int}\,C)C)$ $\Rightarrow^2 (+\underline{int}(-\underline{int}\,\underline{int})C) \Rightarrow^1 (+\underline{int}(-\underline{int}\,\underline{int})(OA)) \Rightarrow^5 (+\underline{int}(-\underline{int}\,\underline{int})(*A)) \Rightarrow^{7,8}$ $(+\underline{int}(-\underline{int}\,\underline{int})(*CCC)) \Rightarrow^{2(\text{tres veces})} (+\underline{int}(-\underline{int}\,\underline{int})(*\underline{int}\,\underline{int}\,\underline{int}))$.

(b) $M = (K, \Sigma, \Gamma, \Delta, q_0, F)$, donde el conjunto de los estados es $K = \{q_0, f\}$, el vocabulario de la cinta es $\Sigma = \{\underline{int}, (,), +, -, *, /\}$, el vocabulario de la pila es $\Gamma = \Sigma \cup V$ siendo $V = \{C, O, A\}$, el estado inicial es q_0, el único estado aceptador es f y Δ está formada por las siguientes transiciones:

1. $((q_0, \lambda, \lambda), (f, C))$.
2. $((f, \lambda, C), (f, (OA)))$.
3. $((f, \lambda, C), (f, \underline{int}))$.
4. $((f, \lambda, O), (f, +))$.
5. $((f, \lambda, O), (f, -))$.
6. $((f, \lambda, O), (f, *))$.
7. $((f, \lambda, O), (f, /))$.
8. $((f, \lambda, A), (f, AC))$.
9. $((f, \lambda, A), (f, C))$.
10. $((f, \underline{int}, \underline{int}), (f, \lambda))$.
11. $((f, (, (), (f, \lambda))$.
12. $((f,),)), (f, \lambda))$.
13. $((f, +, +), (f, \lambda))$.
14. $((f, -, -), (f, \lambda))$.
15. $((f, *, *), (f, \lambda))$.
16. $((f, /, /), (f, \lambda))$.

(c) Cómputo que reconoce $(+\ \underline{int}\ (-\ \underline{int}\ \underline{int}))$:

estado	cinta	pila	transición
q_0	$(+\ \underline{int}\ (-\ \underline{int}\ \underline{int}))$	λ	–
f	$(+\ \underline{int}\ (-\ \underline{int}\ \underline{int}))$	C	1
f	$(+\ \underline{int}\ (-\ \underline{int}\ \underline{int}))$	(OA)	2
f	$+\ \underline{int}\ (-\ \underline{int}\ \underline{int}))$	$OA)$	11
f	$+\ \underline{int}\ (-\ \underline{int}\ \underline{int}))$	$+A)$	4
f	$\underline{int}\ (-\ \underline{int}\ \underline{int}))$	$A)$	13
f	$\underline{int}\ (-\ \underline{int}\ \underline{int}))$	$AC)$	8
f	$\underline{int}\ (-\ \underline{int}\ \underline{int}))$	$CC)$	9
f	$\underline{int}\ (-\ \underline{int}\ \underline{int}))$	$\underline{int}\ C)$	3
f	$(-\ \underline{int}\ \underline{int}))$	$C)$	10
f	$(-\ \underline{int}\ \underline{int}))$	$(OA))$	2
f	$-\ \underline{int}\ \underline{int}))$	$OA))$	11
f	$-\ \underline{int}\ \underline{int}))$	$-A))$	5
f	$\underline{int}\ \underline{int}))$	$A))$	14
f	$\underline{int}\ \underline{int}))$	$AC))$	8
f	$\underline{int}\ \underline{int}))$	$CC))$	9
f	$\underline{int}\ \underline{int}))$	$\underline{int}\ C)$	3
f	$\underline{int}))$	$C))$	10
f	$\underline{int}))$	$\underline{int}))$	3
f	$))$	$))$	10
f	$)$	$)$	12
f	λ	λ	12

(d) Aplicando la regla de recursión, reemplazamos las producciones $A \longrightarrow AC$ y $A \longrightarrow C$ por las producciones $A \longrightarrow CD$, $D \longrightarrow CD$ y $D \longrightarrow \lambda$. Obtenemos entonces la siguiente gramática LL(1) G' equivalente a G:

1. $C \longrightarrow (OA)$.
2. $C \longrightarrow \underline{int}$.
3. $O \longrightarrow +$.
4. $O \longrightarrow -$.
5. $O \longrightarrow *$.
6. $O \longrightarrow /$.
7. $A \longrightarrow CD$.
8. $D \longrightarrow CD$.
9. $D \longrightarrow \lambda$.

(e) La tabla de análisis de G' es la siguiente:

	t	(	)	+	-	*	/
C	2	1					
O				3	4	5	6
A	7	7					
D	8	8	9				

Obsérvese que de la derivación

$C \Rightarrow^1 (OA) \Rightarrow^7 (OCD)$

se deduce que) $\in$ Siguientes(D) y, por tanto, la producción 9 $\in$ TABLA(D,)).