



Apellidos, Nombre.....

--	--	--	--	--	--	--	--	--	--	--

- 1 (2 p.) Dado el autómata siguiente, obtener un R.F.D. mínimo equivalente, el lenguaje que reconoce, y una gramática de tipo 3 para tal lenguaje. Descríbase su complementario.

	a	b	c	ε
→ 1		2	3	2, 3
2	2	3	1, 2	
(3)				

- 2 (0'75 p.) Probar que $L_1 = \{x1^n/x \in (a|b)^*, n = |x|\}$ es independiente de contexto no regular.
- 3 (1'5 p.) Sea $L_2 = \{xw/x \in (a|b)^*, w \in 1(0|1)^*|0, w \text{ representa en binario } |x|\}$
1. Mostrar las cadenas de L_2 de longitud ≤ 4 .
 2. Caracterizar las cadenas de $L_2 \cap (a|b)^*10^*$
 3. Probar que L_2 no es independiente de contexto.
 4. ¿Es L_2 recursivo? Razónese en términos de máquinas de Turing. (Si se plantea una máquina de Turing, no es necesario construirla, pero sí indicar cómo podría hacerse).
- 4 (1'25 p.) Se quiere implantar un análisis sintáctico predictivo basado en la gramática siguiente:

$$G: \begin{cases} S \rightarrow TL ; | LTCB | \epsilon \\ T \rightarrow i | f \\ L \rightarrow L, V | V \\ B \rightarrow i | CB \end{cases} \quad \begin{cases} V \rightarrow id | id [n R] M \\ R \rightarrow , n R | \epsilon \\ C \rightarrow CD \\ M \rightarrow [n] M | \epsilon \end{cases}$$

Realizar las tareas adecuadas para hacerlo posible. Calcular la TASP correspondiente especificando los conjuntos Primeros y Siguietes necesarios para este cálculo. ¿Puede hacerse el análisis deseado?.

- 5 (0'5 p.) Dado un alfabeto Σ y una familia de lenguajes recursivamente numerables $L_1, L_2 \dots L_n$ definidos sobre dicho alfabeto, que constituye una partición de Σ^* , demuéstrase que cada uno de los lenguajes de esta familia es recursivo.
- 6 (1 p.) Respóndase en esta misma página a las siguientes preguntas. Las respuestas correctas se calificarán con 0'2 puntos, las incorrectas con -0'2 puntos y las no contestadas con 0 puntos.
1. Todo autómata a pila es equivalente a un autómata a pila determinista. ¿Cierto o Falso?
 2. Si una gramática no es $LL(1)$, entonces es ambigua. ¿Cierto o falso?
 3. Si una gramática no es ambigua, Yacc lo detectará porque no hallará conflictos. ¿Cierto o falso?
 4. Si un lenguaje es recursivo, entonces es recursivamente numerable. ¿Cierto o falso?
 5. Para cualquier lenguaje existirá una gramática no restringida, más o menos difícil de encontrar, que lo genere. ¿Cierto o falso?

6 Alternativo

1. Todo autómata a pila es equivalente a un autómata a pila determinista. ¿Cierto o Falso?
2. Si un lenguaje es recursivo, entonces es recursivamente numerable. ¿Cierto o falso?
3. Para cualquier lenguaje existirá una gramática no restringida, más o menos difícil de encontrar, que lo genere. ¿Cierto o falso?
4. Si una gramática no es $LALR(1)$, entonces es ambigua. ¿Cierto o falso?
5. Si una gramática no es ambigua, la TASP tendrá a lo sumo una regla por casilla. ¿Cierto o falso?

7 (1 p.) Respóndase a la vuelta de la página¹ a las siguientes preguntas.

1. ¿Para qué sirve el algoritmo CYK?
2. ¿Cómo debe ser la gramática de entrada para el algoritmo CYK?
3. ¿En qué fases se descompone habitualmente el proceso de compilación?
4. ¿Qué característica debe cumplir una máquina de Mealy para ser considerada de Moore?
5. ¿Qué propiedades debe cumplir la función de transición de un autómata a pila para ser determinista?

8 (2 p.) Una persona almacena la información referente a sus facturas de luz, agua y teléfono en un fichero de texto a razón de una factura por línea.

Las facturas de agua están en líneas que comienzan por una letra A en las que aparecen, en este orden y separados por blancos, el nombre de la empresa emisora de la factura, el precio de cada litro de agua y el número de litros consumidos.

Las facturas de luz están en líneas que comienzan por una letra L en las que aparecen, en este orden y separados por blancos, el nombre de la empresa emisora de la factura, el tipo de horario contratado, el precio de cada kw y el número de kw consumidos. El tipo de horario contratado está indicado por una letra D (diurno) o N (nocturno). Si el contrato es nocturno se aplica un descuento del 20 %.

Las facturas de teléfono están en líneas que comienzan por una letra T en las que aparecen, en este orden y separados por blancos, el nombre de la empresa emisora de la factura, el precio del establecimiento de llamada, el número de llamadas, el precio de cada minuto, y el número de minutos consumidos.

Los nombres de todas las empresas consisten en una única secuencia de letras y tienen más de un carácter.

El fichero termina con una línea que sólo contiene la cadena "FIN".

En el fichero pueden aparecer varias facturas del mismo tipo, pero en ese caso tendrán que ser de empresas diferentes; es decir: el contenido del fichero es de una línea por cada empresa.

Usando las herramientas Lex y Yacc, construir un programa que, tomando como entrada un fichero con este formato (se supone que se redirige la entrada, no es necesario pasar el nombre de fichero como argumento), escriba en pantalla una línea por cada factura, en la que aparezca el nombre de la empresa emisora de esa factura y el total que hay que pagarle. Además, si el fichero sigue todas las especificaciones mencionadas, el programa debe escribir en pantalla el mensaje "FICHERO DE ENTRADA CORRECTO".

Por ejemplo, dado el fichero de entrada

```
T Amena 0.65 7 0.3 25
A AguasValladolid 0.1 109
T Telefonica 0.63 12 0.35 78
L UnionFenosa N 0.56 150
L Endesa D 0.48 60
FIN
```

la salida (los comentarios no salen) debería ser:

```
Amena 12.05 // 12.05 = 0.65 * 7 + 0.3 * 25
AguasValladolid 10.9 // 10.9 = 0.1 * 109
Telefonica 34.86 // 34.86 = 0.63 * 12 + 0.35 * 78
UnionFenosa 67.2 // 67.2 = 0.56 * 150 - 20%
Endesa 28.8 // 28.8 = 0.48 * 60
FICHERO DE ENTRADA CORRECTO
```

¹la vuelta de la página estaba en blanco en el enunciado original