

Apellidos, Nombre: _____

D.N.I., Firma: _____

1	2	3	4	5	6	7	8	9	10	Suma
(20)	(15)	(10)	(5)	(5)	(5)	(5)	(5)	(18)	(12)	(100)

El examen consta de tres partes:

1. Las preguntas de la primera parte (preguntas 1–3) se contestan en hojas adicionales a gusto del estudiante.
2. Las preguntas de la segunda parte (preguntas 4–8) se deben contestar con un simple **si** o **no**, razonando después la respuesta brevemente en el espacio disponible en la hoja para tal fin.
3. Las preguntas de la tercera parte (prácticas, preguntas 9–10) se contestan en hojas adicionales a gusto del estudiante.

Tiempo para realizar el examen:

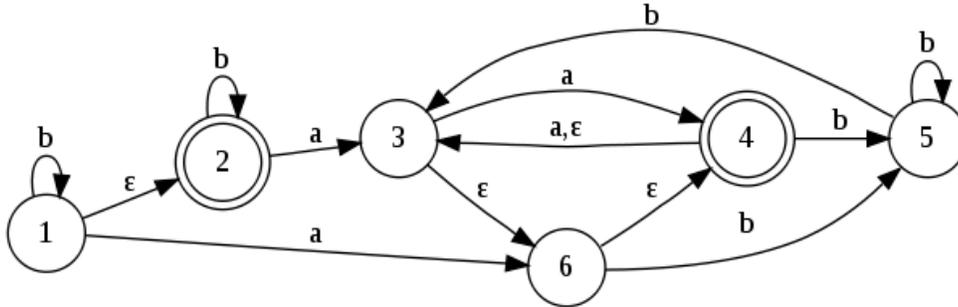
1. La duración del examen es de **3 horas**.

Nota: Se tiene que aprobar la parte de teoría (llegar a 35 puntos), la parte de prácticas se suma. Se aprueba la asignatura, si la suma es por lo menos 50 puntos.

Primera parte

Pregunta 1: [20 Puntos]

Calcula el $Indice(R_{\overline{L}})$, es decir, del complemento del lenguaje L , que acepta el siguiente AFND- ϵ :



Pregunta 2: [15 Puntos]

Construye una gramática en forma normal de Chomsky G que genera el lenguaje

$$L = \{a^i b^j c^k d^l \mid i = k \text{ ó } j = l \text{ con } i, k \geq 0 \text{ y } j, l > 0\} \cup \epsilon$$

ejemplos de palabras que pertenecen a L :

$$\epsilon, aaabbcccd, abcd, aaaaaabbcccd, bbbddd$$

Pregunta 3: [10 Puntos]

Construye una gramática libre de contexto que no sea regular. Argumenta por qué el lenguaje generado por la gramática de tu solución:

1. es libre de contexto,
2. no es regular.

Segunda parte

Pregunta 4: [5 Puntos]

Dado un lenguaje L que es libre de contexto. ¿El complemento de L , es decir, \overline{L} puede ser libre de contexto (es decir, existe algún L con tal propiedad)?

Pregunta 5: [5 Puntos]

Sea $L_1 \subset \Sigma^*$ algún lenguaje *no* libre de contexto. ¿Existe un lenguaje L_2 de tal manera que para la intersección $L_1 \cap L_2$ se cumpla $Indice(R_{L_1 \cap L_2}) = 1$?

Pregunta 6: [5 Puntos]

¿Si un autómata finito determinista completo tiene por lo menos dos estados no-finales accesibles de los cuales uno tiene una arista reflexiva pero el otro no, entonces dichos estados pueden ser equivalentes?

Pregunta 7: [5 Puntos]

¿Es posible averiguar si un autómata finito con pila no-determinista (AFPND) M que acepta en estado final acepta la palabra vacía, es decir, que $\epsilon \in L(M)$?

Pregunta 8: [5 Puntos]

Sean α y β dos expresiones regulares sobre el alfabeto $\{a, b\}$. ¿Es posible construir una gramática lineal por la derecha G de tal manera que $L(G) = \overline{L(\alpha)} \cup \overline{L(\beta)}$?

Tercera parte (Prácticas)

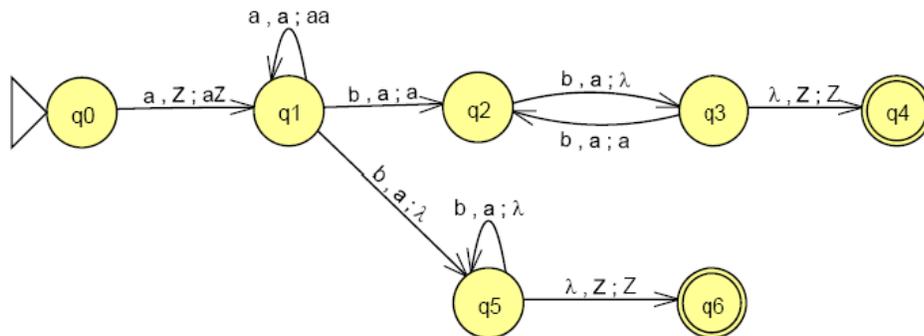
Pregunta 9: [18 Puntos]

Acerca de la herramienta (JFLAP) usada en prácticas (18 puntos):

1. ¿Se puede transformar una gramática regular a una expresión regular en cualquier caso?
2. ¿En qué formato guarda los ficheros (XML, Texto plano, binario...)?
3. ¿Qué algoritmo usa para la minimización de autómatas?
4. ¿Cómo podemos insertar el símbolo de la palabra vacía (ϵ ó λ) en el editor de expresiones regulares?
5. ¿Cómo hacemos para indicar que una transición entre dos estados acepta varios símbolos?
6. ¿En la edición de una gramática, cuáles son los caracteres válidos para símbolos terminales y no terminales?
7. En la edición de una gramática como denotamos la variable de comienzo?
8. Teniendo dos autómatas, uno determinista y el otro no. ¿Se puede comprobar directamente si dos autómatas son equivalentes? Indicar el procedimiento, o en su caso las operaciones intermedias que hay que hacer.
9. ¿De que manera comprobamos si una cadena pertenece al lenguaje generado por una gramática? ¿Es siempre posible determinarlo?
10. Explica que función tienen los tres elementos en una transición de un autómata de pila.
11. ¿Cuál es el proceso para transformar una expresión regular a una gramática equivalente?
12. Indica las condiciones que se tienen que dar para que un autómata de pila valide una palabra en JFLAP.

Pregunta 10: [12 Puntos]

Describe todos los posibles “traces” (o cálculos) del siguiente autómata con las cadenas: ϵ , ab , aab , abb , bb , $aabb$.



Determina si el autómata acepta o no acepta cada una de las palabras.