

**112034 Lenguajes y Autómatas** Trimestre 19P.

Docente: Dr. Carlos Barrón Romero

Lista de Ejercicios para el Examen global de Lenguajes y Autómatas.

Instrucciones. El marco de sus respuestas son los objetivos de la UEA que transcribo a continuación:

- Describir, interpretar e ilustrar los modelos teóricos de cómputo.
- Describir los conceptos de lenguaje formal y gramática.
- Reconocer y diferenciar las clases de lenguajes formales asociadas con cada modelo teórico de cómputo.

Responda en forma resumida, que su respuesta refleje los objetivos de la UEA, use el sentido común y describa con claridad la explicación o el desarrollo de su solución. El valor de cada pregunta está entre "[", "]"".

1. Sea el conjunto  $\mathbb{Q}$  el conjunto de los números racionales de la forma  $p/q$  donde  $p \in \mathbb{Z}$ ,  $q \in \mathbb{N} \setminus \{0\}$  y  $\Sigma = \{0, 1, +, -, /\}$ 
  - (a) Construir un autómata finito determinístico (AFD) que reconozca a  $\mathbb{Q}$  usando  $\Sigma$ .
  - (b) Construir una expresión regular (ER) para  $\mathbb{Q}$  usando el formato descrito y  $\Sigma$ .
2. Calcular al menos 5 elementos de los lenguajes.
  - (a)  $\Sigma = \{a, b, c, d, e\}$ .  $L = \{x \in \Sigma^* \mid x \text{ tiene al menos una sílaba (o sea: una consonante seguida de una vocal.)}\}$ .
  - (b)  $\Sigma = \{0, 1\}$ .  $L = \{x \in \Sigma^* \mid x \text{ se interpreta como valores } 2^k + 1, k \in \mathbb{N}\}$ .
3. Sea  $\Sigma = \{a, b\}$ .
  - (a) Construir un autómata de pila (ADP) para un lenguaje sobre  $\Sigma$  que no corresponda con una expresión regular. Explicar porque su lenguaje propuesto no lo reconoce un autómata finito determinístico.
  - (b) Mostrar las derivaciones de reconocimiento de su ADP con dos ejemplos.
4. Construir una gramática para el lenguaje  $L = \{a, a+a, a+a+a, a+a+a+a, \dots\}$ .
5. Construir una Máquina de Turing (MT) apropiada para multiplicar por 3 a un número positivo ( $3*4=12$ ). Notas: Definir una convención de representación de números positivos que le convenga y puede usar en la construcción de su MT, Máquinas de Turing del libro o sus apuntes.
  - (a) Mostrar como funciona su MT para 0, 1 y 2.
6. Explicar y dar un ejemplo de un lenguaje que no sea reconocido por un Autómata Finito Determinístico (AFD), pero que sea reconocido por una MT.
7. Sea  $\Sigma = \{a, b, c\}$ .
  - (a) Construir un Lenguaje Recursivo sobre  $\Sigma$ . Es decir describa una MT apropiada.
  - (b) Mostrar los dos casos de las derivaciones de reconocimiento de su lenguaje con ejemplos.
8. Explicar con un ejemplo como se codifica una MT para una Máquina de Turing Universal.
9. Sea  $\Sigma = \{0, 1\}$ . Explicar si es falso o verdadero cada uno de los incisos siguientes:
  - (a)  $2 \in \Sigma$ .
  - (b)  $00 \in \Sigma$ .
  - (c)  $\Sigma \subset \Sigma^*$
  - (d)  $\{0, 1\} \in 2^\Sigma$ , donde  $2^\Sigma$  es el conjunto potencia de  $\Sigma$ .
  - (e)  $2^\Sigma$  es un lenguaje sobre  $\Sigma$ .
10. Sea  $\Sigma = \{a, b, 0, 1\}$ . Escribir 5 palabras diferentes de los lenguajes.
  - (a)  $L_1 = \{x \in \Sigma^* \mid x \text{ tiene de prefijo una vocal y al menos un dígito}\}$ .

(b)  $(\mathbf{1}^* + \mathbf{0+a}) \mathbf{b}^*$ .

11. Sea  $\Sigma = \{u, z, 0, 1\}$  y  $L_1 = \{x \in \Sigma^* \mid x \text{ tiene de prefijo una vocal y al menos un dígito}\}$

(a) Construir una ER para  $L_1$ , en caso de no poder explicar.

(b) Construir un  $AFN - \varepsilon$  para  $L_1$ , en caso de no poder explicar.

12. Explicar la relación de los lenguajes regulares, o sea entre los lenguajes derivados de las Expresiones Regulares y de los autómatas AFD, AFN, AFN- $\varepsilon$ .

13. Sea  $\Sigma = \{z, 1\}$  y  $L_z = \{z^k 1 z z^k \mid k = 1, 2, 3, \dots\}$ .

(a) Escribir 5 palabras de  $L_z$ .

(b) Construir una gramática para  $L_z$ . Dar y mostrar con dos ejemplos de derivación que funciona su gramática.

(c) Construir un autómata de pila ( $AP$ ) para  $L_1$ . Dar y mostrar con dos ejemplos de derivación que funciona su  $AP$ .

14. Explicar o dar un ejemplo de un lenguaje regular que no pueda ser reconocido por un  $AP$ .

15. Sea  $\Sigma_1 = \{e, 1\}$ .

(a) Construir un lenguaje no regular sobre  $\Sigma_1$ .

(b) Construir una MT para generar las palabras de su lenguaje del inciso anterior.

16. Sea  $\Sigma = \{0, 1\}$ . Establecer una correspondencia entre lenguajes de los siguientes incisos y el autómata finito determinístico (AFD), el autómata de pila ( $AP$ ) y la Máquina de Turing (MT) y Lenguajes Regulares (LR), Lenguajes Independientes del Contexto (LIC), Lenguajes Dependientes del Contexto (LDC) y lenguajes Libres (LL). Escribir en el espacio subrayado todas las abreviaciones que correspondan con cada lenguaje con su maquina o estructura más apropiada para reconocerlo o generarlo.

(a)  $\mathbf{1}^*$  : .....

(b)  $\{0^k 1 0^k \mid k = 1, 2, 3, \dots\}$  : .....

(c)  $\{0^k 1^k 0^k \mid k = 1, 2, 3, \dots\}$  : .....

**Nota.** Dada cualquier  $x \in \Sigma^*$  y cualquier  $MT = (q_0, \Sigma, Q, \delta, F)$  se tienen 3 posibilidades:

1)  $x \in L(MT)$

2)  $x \notin L(MT)$

3) Con la entrada  $x$  la  $MT$  no se detiene, i.e.,  $x$  no es computable por la  $MT$ . (Note que dado  $x \in \Sigma^*$ ,  $x$  es computable por una  $MT$  significa que tal  $MT$  se detiene con la entrada  $x$ ).

Definición Un Lenguaje Recursivo o Lenguaje Computable es aquel que dada una  $MT$  cumple 1 y 2.

Definición Un Lenguaje Recursivo Numerable es aquel que dada una  $MT$  cumple 1, 2 y 3.

1. Explicar con ejemplos el funcionamiento de las funciones recursivas y sus MT.

2. 1) Explicar si la serie de números conocidos como serie de Fibonacci se pueden construir con funciones recursivas. Luego, 2) explicar si el lenguaje de los números de Fibonacci son un lenguaje Computable. La fórmula de los números de Fibonacci se define con las fórmulas 1)  $f_0 = 0, f_1 = 1$  2) [Fórmula inductiva]  $f_{n+1} = f_n + f_{n-1}$  con  $n \geq 1, n \in \mathbb{N}$ . Es decir, 2) consiste en responder si  $L_F = \{x \in \mathbb{N} \mid x \text{ es un número de Fibonacci}\}$  es un Lenguaje Recursivo o Computable.

3. Explicar porqué dada una función recursiva siempre se puede construir un Lenguaje Computable.

4. Construir un lenguaje Computable sobre  $\Sigma = \{0, 1\}$ . Es decir describa una MT apropiada. Mostrar los casos de las derivaciones de reconocimiento de su lenguaje con ejemplos.

5. Explicar la relación entre las clases de los lenguajes (Regulares, independientes del contexto, dependientes del contexto y libres) derivados de una estructura gramatical o de un AFD, de un automata de pila y de una Máquina de Turing.

6. Explicar si es cierto que el complemento de un lenguaje recursivo numerable es recursivo numerable. Es decir, construir una Máquina de Turing (MT) apropiada para el complemento del lenguaje Recursivo Enumerable dado y después mostrar como funciona su MT cuando 1)  $x \in L(MT)$ , 2)  $x \notin L(MT)$  y 3)  $MT$  no se detiene para  $x$ .
7. Explicar si es cierto que el complemento de un lenguaje computable es computable. Es decir, construir una Máquina de Turing (MT) apropiada para el complemento del lenguaje Computable dado y después mostrar como funciona su MT cuando 1)  $x \in L(MT)$  y 2)  $x \notin L(MT)$  [Explicar porqué no se cumple 3)  $MT$  no se detiene para  $x$ , en este caso de un lenguaje computable].

NOTA: Para el examen global se incluyen las listas de ejercicios de los parciales.