



MÁQUINAS DE TURING

1. Construir una máquina de Turing de tres pistas que reste el número binario de la segunda pista del número binario de la primera y deje el resultado en la tercera pista con las siguientes especificaciones:
 - a) En la primera y segunda cinta inicialmente sólo hay caracteres en blanco salvo sendas tiras de ceros y unos.
 - b) Tanto el minuendo como el substraendo expresan números naturales codificados en binario natural de la misma longitud, los cuales se encuentran alineados.
 - c) La tercera cinta inicialmente está llena de caracteres en blanco.
 - d) El resultado deberá estar alineado por la derecha con los datos de las otras dos cinta, de tal manera, que si el resultado es negativo, deberá parecer un “1” en la cifra más significativa, considerándose que lo que viene a continuación es el valor absoluto de la diferencia.
 - e) Inicialmente el cabezal puede estar situado sobre cualquier par de dígitos de los datos de entrada. (junio 1996)
2. Construir una máquina de Turing que acepte el lenguaje: $L = \{w \in \Sigma / |w|_a = |w|_b\}$; donde el alfabeto es: $\Sigma = \{a, b, c, d, e\}$ (septiembre 1996)
3. Dada la gramática independiente del contexto:
 $S \rightarrow aSB/\lambda$
 $B \rightarrow b/bb$ donde a y b son símbolos terminales. Por su parte B es un auxiliar y S el símbolo inicial.
Obtener:
 - El lenguaje L que representa.
 - Un autómata de pila no determinista que acepte a L.
 - Una máquina de Turing con una sola cinta que reconozca a L. Asimismo, mostrar la sucesión de descripciones instantáneas por las que atravesaría el autómata al procesar dos cadenas de longitud cinco: una de ellas aceptada y la otra no. (junio 1997)
4. Probar que si L es un lenguaje regular, existe una máquina de Turing, que reconoce a todas las cadenas w pertenecientes a dicho lenguaje en un número de pasos no superior a $|w|+1$. (sept. 1997)
5. Diseñar una máquina de Turing con tan sólo tres estados, que sea capaz de generar el código binario natural en orden creciente. La situación de partida corresponde con una cinta repleta de caracteres

blanco salvo un cero, al que precisamente apunta el cabezal antes de comenzar a funcionar. Lógicamente, este autómata no se para nunca. Dispondrá de estados finales, en los cuales se realizaría la lectura de la cinta. Así, se irían registrando salidas: b0b, b1b, b10b, b11b, b100b, b101b, b110b, b111b, etc. Cuando no se encuentre en uno de estos estados de aceptación, la cinta se utilizará como se estime más oportuno. (junio 1998)

6. Construir una máquina de Turing con tres pistas, que se pare en un estado final solamente si el número binario almacenado en la primera pista es un múltiplo del de la segunda, también codificado en binario natural. Se considera estos dos números alineados por el bit menos significativo, posición a la que apunta el cabezal en el instante inicial. Respecto a la tercera pista, se supondrá que arranca vacía de contenido (espacios en blanco). (enero 1999)
7. Dado el siguiente lenguaje $L = \{a^m b^n a^m b^n / n, m > 0\}$, estudiar su pertenencia al conjunto de los independientes del contexto. En caso afirmativo, obtener el autómata de pila no determinista que lo reconoce. En caso negativo, diseñar una máquina de Turing que acepte dicho lenguaje. (ene. 2000)
8. Diseñar una máquina de Turing con una sola cinta que produzca en orden creciente a su longitud las cadenas de $L = \{a^{n^2} / n \in \mathbb{N}\}$. Inicialmente, dicho autómata arranca con toda la cinta llena de espacios en blanco, la cual se podrá utilizar como se estime más conveniente, pero se ha de reservar una ventana para mostrar la cadena generada. Este trozo de cinta estará acotado por espacios en blanco, en medio de los cuales se situará la palabra correspondiente, cuya lectura se realizará cada vez que se alcance el estado final q_f . En este instante, el cabezal deberá estar en el extremo inmediatamente anterior a la palabra, es decir, en el primer blanco a la izquierda. Para facilitar la construcción, ocasionalmente se permitirá la parada del cabezal en alguna transición. (sept. 2000)
9. Pruébese que el lenguaje $L = \{a^i / i = 2^n \quad \forall n \in \mathbb{N}\}$ es recursivo. (junio 2001)
10. Diseñese una máquina de Turing, con no más de cinco estados, que acepte el lenguaje, dentro del alfabeto $\{a, b, c\}$, cuyas cadenas no tienen ninguna "a" en su primera mitad y ninguna "b" en su segunda. A tenor de esta especificación, debe entenderse que las palabras de longitud impar no son admitidas. (septiembre 2002)
11. Se dispone de un programa de ordenador concebido para reconocer un determinado lenguaje, que es recursivamente enumerable y no recursivo. ¿Cómo se comportará este programa ante una cadena arbitraria de entrada? Escoja una de las siguientes respuestas y justifíquese su elección.
 - a) no parará.
 - b) parará respondiendo afirmativa o negativamente a la aceptación de la cadena como parte del lenguaje.
 - c) se detendrá aceptando a todas las cadenas del lenguaje, pero es posible que para alguna del complementario no se pare.
 - d) ante todas las cadenas del lenguaje se parará respondiendo afirmativamente, pero siempre habrá una cadena del complementario, al menos, para la cual dicho programa no se detendrá.

Análogamente, imagínese un "superprograma" al que se le introduce un programa arbitrario y cualquiera de sus entradas. La misión de este "superprograma" es determinar de antemano si el programa suministrado, ante una entrada particular, se detendrá o no. ¿Qué se puede comentar

19. Estúdiense la pertenencia a los lenguajes recursivos de:

$$L = \{w_1\#w_2\#\dots\#w_n; w_i \in (0|1)^*; i = 1,2,\dots,n / \exists j = 1,2,\dots,n \quad j)_2 = w_j\}$$

Como puede verse, cada palabra del lenguaje se compone de una serie de subcadenas separadas por el símbolo # y numeradas por su orden de aparición (j) de izquierda a derecha. Si dichas subcadenas se interpretaran como un número codificado en binario natural, bastaría con que uno de estos valores coincidiera con el de su orden j, para que la palabra fuera incluida en el lenguaje L.

(septiembre 2007)

20. Dado un programa de ordenador, del que se sabe que en algunos casos entra en bucle infinito, ¿se podría siempre sustituirlo por otro cuya respuesta fuera la misma salvo la de esos bucles infinitos, situación que predeciría? Explíquese detalladamente la respuesta. (enero 2008)
21. Sea la operación $P(L) = \{x \in L/x^l \notin L\}$. Verifíquese si esta operación es cerrada en la clase de los lenguajes recursivos. (junio 2008)
22. Considérese un lenguaje $L \subseteq (a|b)^*$ y el obtenido a partir de él como: $L_c = c^+L \cup (a|b)^*$. Para no dar lugar a dudas, conviene aclarar que el nuevo lenguaje cumple que: $L_c \subseteq (a|b|c)^*$. Compruébese, entonces, si las propiedades de L respecto a una máquina de Turing son las mismas que las de L_c y viceversa. (junio 2008)
23. Demuéstrese que la intersección de dos lenguajes recursivamente enumerables es una operación cerrada. Ahora considere que uno de los lenguajes anteriores no es recursivo y el otro es regular. ¿La intersección da siempre como resultado un lenguaje recursivo? Si la respuesta es negativa, estúdiense los casos particulares donde esto sí se da. (septiembre 2008)
24. Sea la operación f sobre una palabra x definida por:

$$f(x) = \begin{cases} x_1x_2^l; & \text{si } |x| \text{ es par, con } x = x_1x_2, \text{ donde } |x_1| = |x_2| \\ x_1ax_2^l; & \text{si } |x| \text{ es impar, con } x = x_1ax_2, \text{ donde } |x_1| = |x_2| \text{ y } a \in \Sigma \end{cases}$$

Si esta operación se extiende a lenguajes de la manera habitual, esto es, $f(L) = \{f(x)/x \in L\}$, verifíquese si es cerrada en el conjunto de los recursivos. (septiembre 2008)