

Hoja de Problemas 13Máquinas de Turing

NIVEL DEL EJERCICIO : (★) básico, (♣) medio, (♠) avanzado.

1. (★) Describe una máquina de Turing que acepte el siguiente lenguaje:

$$L = \{a^n b^n \mid n > 0\}.$$

Solución:

2. (★) Describe una máquina de Turing que acepte el siguiente lenguaje:

$$L = \{a^n b^n c^n \mid n > 0\}.$$

Solución:

3. (★) Describe una máquina de Turing que acepte el siguiente lenguaje:

$$L = \{x c x \mid x \in \{a, b\}^+\}.$$

Solución:

4. (★) Describe una máquina de Turing que acepte el siguiente lenguaje:

$$L = \{x \in \{a, b, c\}^* \mid n_a(x) = n_b(x) = n_c(x)\}.$$

Solución:

5. (♣) Describe una máquina de Turing que acepte el lenguaje de los palíndromos sobre el alfabeto $\Sigma = \{a, b\}$.

Solución:

6. (★) Describe una máquina de Turing que acepte el lenguaje de las palabras no vacías sobre $\{a, b, c\}$ tal que la primer letra de la palabra no aparece en el resto de la palabra.

Solución:

7. (★) Describe una máquina de Turing que acepte el siguiente lenguaje:

$$L = \{\mathbf{x} \in \{a, b, c\}^* \mid \mathbf{x} = a^n b^m c^r \text{ y } |x| > 0\}.$$

Además, la máquina debe tener cumplir lo siguiente:

- (a) M debe tener un único estado final.
- (b) M arranca en el estado inicial q_0 con la cabeza apuntando sobre el primer símbolo de la entrada.
- (c) Al parar en el estado final, el contenido de la cinta es el mismo que al principio (la palabra), y la cabeza esta apuntando al primer símbolo de la palabra.

Es decir, para cada $\mathbf{x} \in L$, la máquina debe realizar la siguiente serie de movimientos:

$$q_0 \mathbf{x} \vdash^* q_f \mathbf{x} \text{ donde } q_f \text{ es el estado final.}$$

Solución:

8. (★) Describe una máquina de Turing que acepte el siguiente lenguaje:

$$L = \{\mathbf{x} \in \{a, b, c\}^* \mid \mathbf{x} = a^n b^m c^r \text{ y } |x| > 0 \text{ y si } n_a(\mathbf{x}) > 2, \text{ entonces } \mathbf{x} \text{ tiene al menos una } b\}.$$

Utiliza la máquina del ejercicio 7.

Solución:

9. (♣) Diseña una máquina de Turing que reconozca el siguiente lenguaje:

$$L = \{a^n b^{n+m} a^m \mid n, m \geq 0\}$$

Solución:

$$M = (\{a, b\}, \{a, b, \bullet\}, \bullet, \{q_0, \dots, q_7\}, q_0, f, \{q_7\})$$

$$\begin{aligned} f(q_0, \bullet) &= (q_7, \bullet, R) \\ f(q_0, a) &= (q_1, \bullet, R) \\ f(q_0, b) &= (q_3, \bullet, R) \\ f(q_1, a) &= (q_1, a, R) \\ f(q_1, b) &= (q_1, b, R) \\ f(q_1, \bullet) &= (q_2, a, L) \\ f(q_2, a) &= (q_2, a, L) \\ f(q_2, b) &= (q_2, b, L) \\ f(q_2, \bullet) &= (q_0, \bullet, R) \\ f(q_3, a) &= (q_3, a, R) \\ f(q_3, b) &= (q_3, b, R) \\ f(q_3, \bullet) &= (q_4, \bullet, L) \\ f(q_4, a) &= (q_5, \bullet, L) \\ f(q_5, a) &= (q_5, a, L) \\ f(q_5, b) &= (q_5, b, L) \\ f(q_5, \bullet) &= (q_6, \bullet, R) \\ f(q_6, \bullet) &= (q_7, \bullet, R) \\ f(q_6, b) &= (q_3, \bullet, R) \end{aligned}$$

10. (★) Describe una máquina de Turing que calcule la función $f : \mathbb{N} \rightarrow \{1\}$ definida de la siguiente manera:

$$f(\mathbf{x}) = 1 \quad \forall \mathbf{x} \in \mathbb{N}.$$

Elige y describe una codificación para la entrada y la salida.

Solución:

11. (★) Define una máquina de Turing que calcule la función $f : \mathbb{N} \times \mathbb{N} \rightarrow \mathbb{N}$ definida de la siguiente manera:

$$f(\mathbf{x}, \mathbf{y}) = \begin{cases} \mathbf{y} & \text{si } \mathbf{x} > 0 \\ 0 & \text{en otro caso} \end{cases}$$

Elige y describe una codificación para la entrada y la salida.

Solución:

12. (★) Describe una máquina de Turing que calcule la función $f : \{a, b, c\}^+ \rightarrow \mathbb{N}$ definida de la siguiente manera:

$$f(\mathbf{x}) = n_a(\mathbf{x}) \quad \forall \mathbf{x} \in \{a, b, c\}^+.$$

Elige y describe una codificación para la entrada y la salida.

Solución:

13. (★) Describe una máquina de Turing que calcule la función $f : \mathbb{N} \rightarrow \{a, b\}^*$ definida de la siguiente manera:

$$f(n) = a^n b^n \quad \forall n \in \mathbb{N}.$$

Elige y describe una codificación para la entrada y la salida.

Solución:

14. (★) Describe una máquina de Turing binaria (sólo dos símbolos en la cinta) que calcule la función $f : \mathbb{N} \rightarrow \{0, 1\}$ definida de la siguiente manera:

$$f(\mathbf{x}) = \begin{cases} 1 & \text{si } \mathbf{x} \text{ es par} \\ 0 & \text{en otro caso} \end{cases}$$

Elige y describe una codificación para la entrada y la salida.

Solución:

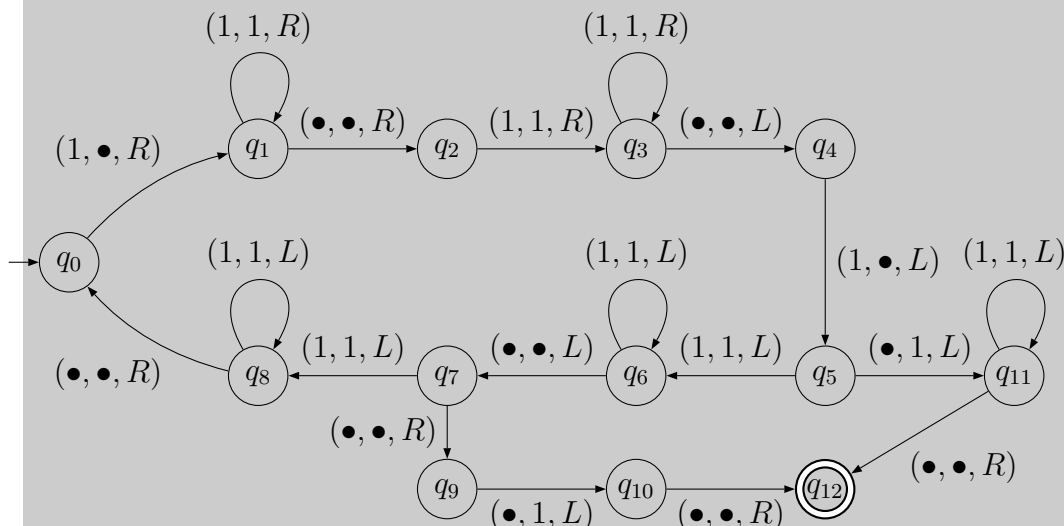
15. (♣) Describe una máquina de Turing binaria (sólo dos símbolos en la cinta) que calcule la función $f : \mathbb{N} \times \mathbb{N} \rightarrow \mathbb{N}$ definida de la siguiente manera:

$$f(\mathbf{x}, \mathbf{y}) = |\mathbf{x} - \mathbf{y}| \quad \forall \mathbf{x}, \mathbf{y} \in \mathbb{N}.$$

Elige y describe una codificación para la entrada y la salida.

Solución:

La entrada y la salida serán codificadas en unario. Los dos parámetros x e y , estarán separados por un blanco. Para computar esta función en una máquina de Turing, tendremos que obtener: $q_0 \mathbf{x} \bullet \mathbf{y} \vdash^* q_f \mathbf{z}$, siendo $\mathbf{z} = f(\mathbf{x}, \mathbf{y})$.



16. (♣) Diseñar una máquina de Turing que implemente la siguiente función:

$$f : \mathbb{N} \times \mathbb{N} \rightarrow \mathbb{N}$$

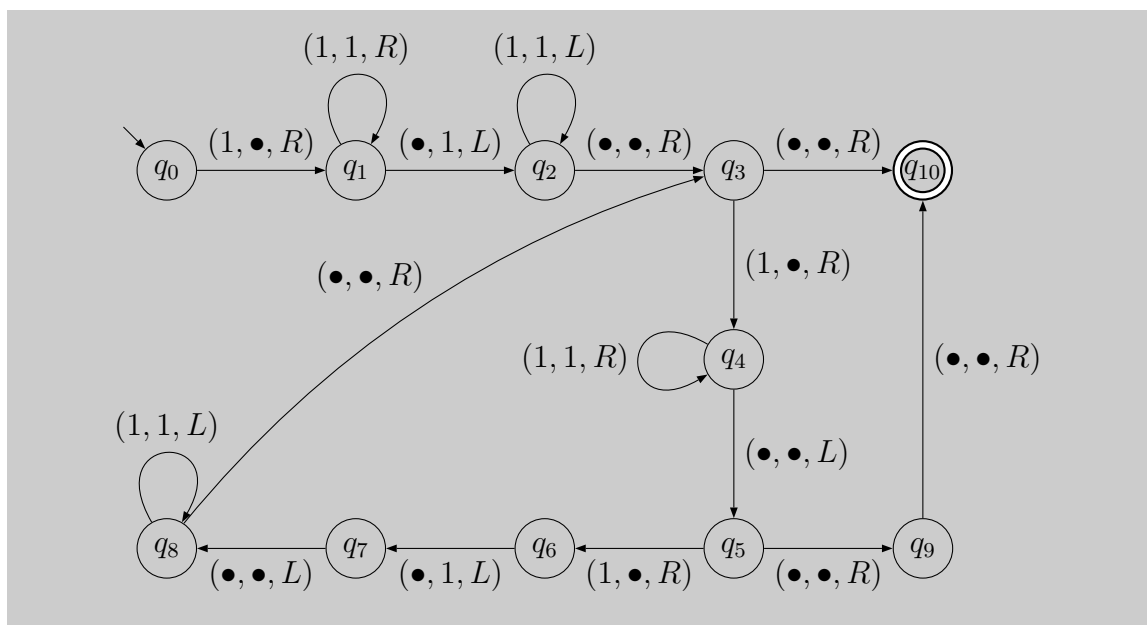
$$f(\mathbf{x}, \mathbf{y}) = \left\lfloor \frac{\mathbf{x} + \mathbf{y}}{2} \right\rfloor \quad \forall \mathbf{x}, \mathbf{y} \in \mathbb{N}.$$

Los parámetros y la solución deberán codificarse en unario; los parámetros de entrada se encontrarán separados por el símbolo blanco. Explique brevemente el algoritmo implementado.

Solución:

$$M = (\{1, \bullet\}, \{1\}, \bullet, \{q_0, q_1 \dots q_{10}\}, q_0, f, \{q_{10}\})$$

Donde la función de transición f , se define de la siguiente manera:



17. (♣) Describe una máquina de Turing, y explica brevemente su funcionamiento, que calcule la función multiplicación de dos números naturales que se define de la siguiente manera:

$$f : \mathbb{N} \times \mathbb{N} \rightarrow \mathbb{N}$$

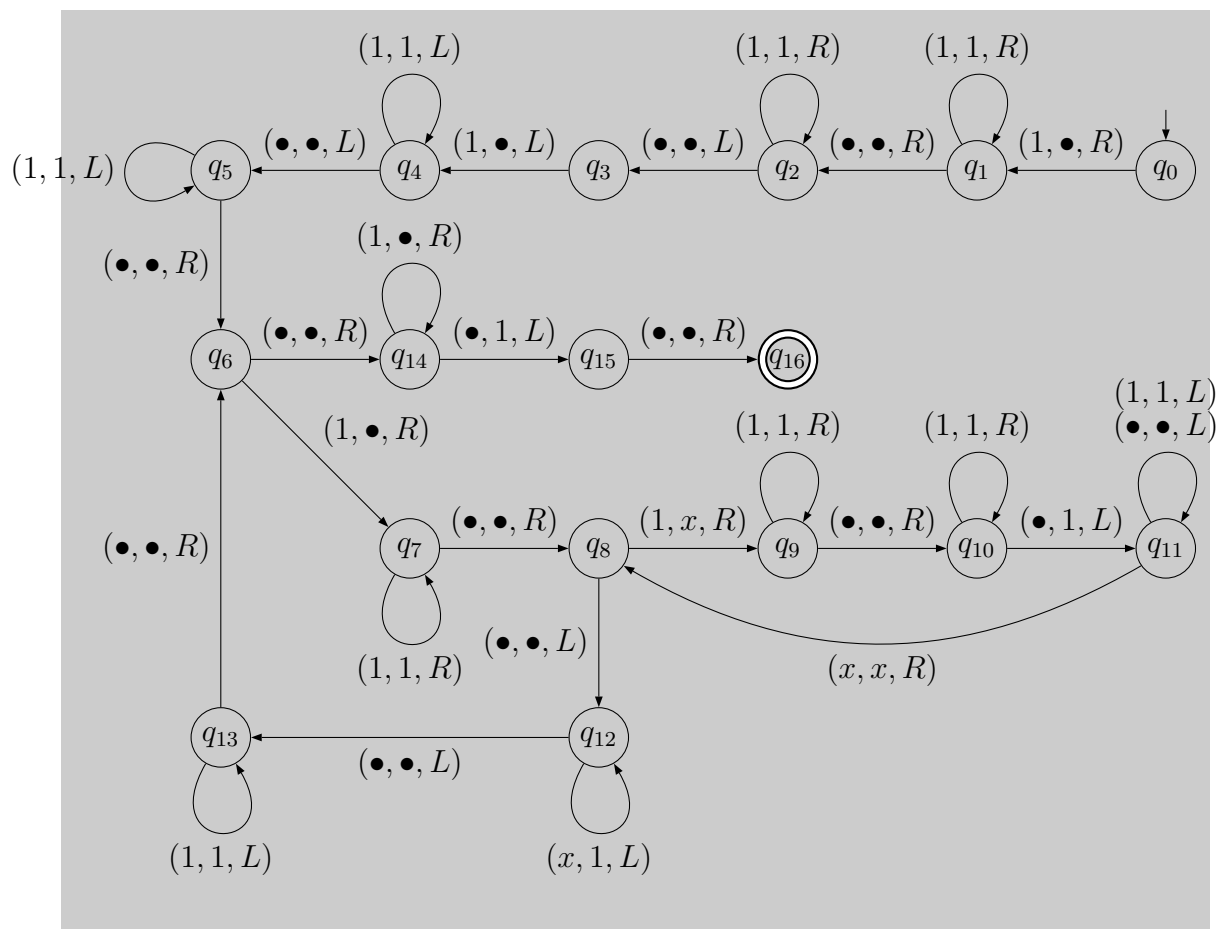
$$f(\mathbf{x}, \mathbf{y}) = \mathbf{x} * \mathbf{y} \quad \forall \mathbf{x}, \mathbf{y} \in \mathbb{N}.$$

Los parámetros y la solución estarán codificados en **unario** y los dos parámetros se encuentran separados por un carácter en blanco.

Solución:

$$M = (\{1, x, \bullet\}, \{1\}, \bullet, \{q_0, q_1 \dots q_{16}\}, q_0, f, \{q_{16}\})$$

Donde la función de transición f , se define de la siguiente manera:



18. (♣) Diseñe una máquina de Turing que implemente la función “potencia de la palabra de un lenguaje” y **explique el algoritmo implementado**. Recuerde que la “potencia de la palabra de un lenguaje” se define como:

Sea \mathbf{x} un número natural, e \mathbf{y} una palabra. La potencia i -ésima de \mathbf{y} , denominada $\mathbf{y}^{\mathbf{x}}$, es la operación que consiste en concatenarla consigo misma \mathbf{x} veces.

La función se define como:

$$f(\mathbf{x}, \mathbf{y}) = \mathbf{y}^{\mathbf{x}} \quad \text{donde : } \mathbf{y} \in \{a, b\}^*, \quad |\mathbf{y}| \geq 1, \quad \mathbf{x} \in \mathbb{N}.$$

Tenga en cuenta que:

- El parámetro \mathbf{x} deberá codificarse en unario.
- Los parámetros de entrada se encontrarán separados por el símbolo blanco.
- El primer parámetro será el exponente (\mathbf{x}) y el segundo la palabra (\mathbf{y}).

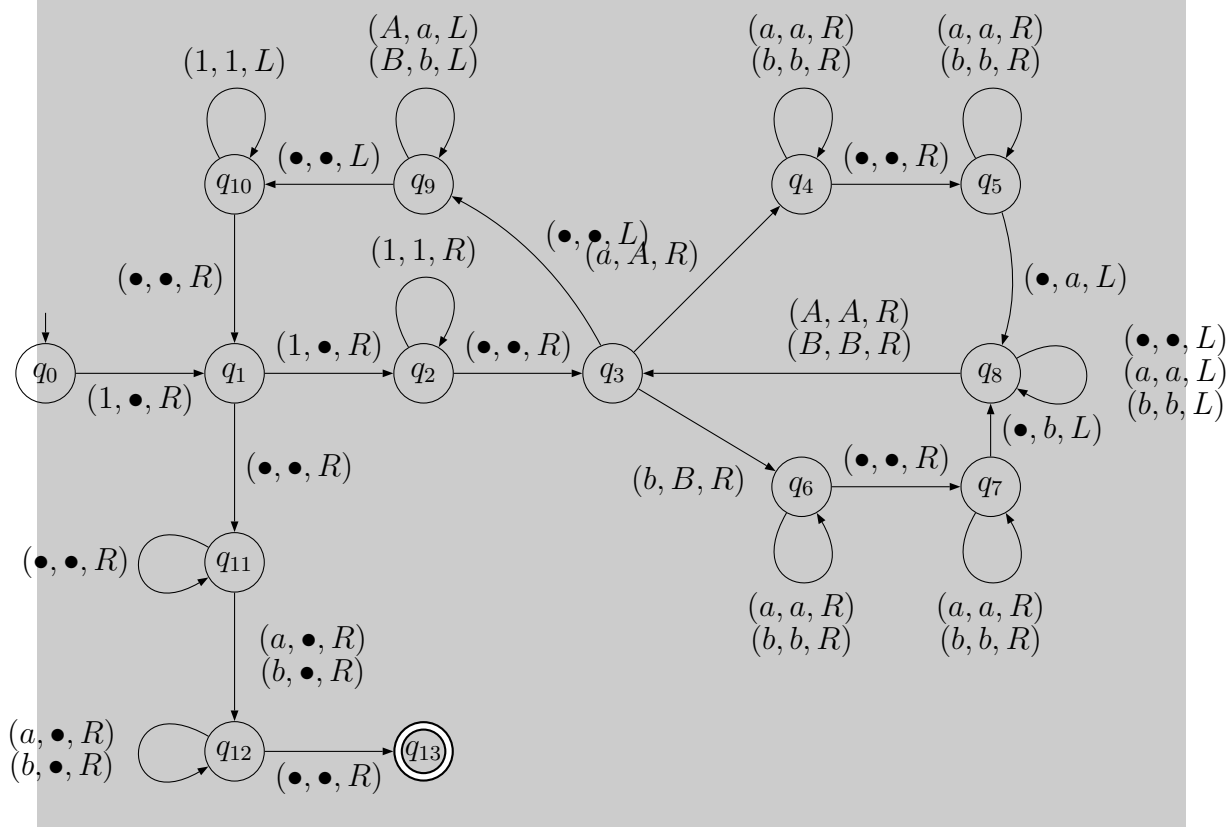
Ejemplo de uso:

$$\bullet q_0 111 \bullet ab \bullet \vdash^* \bullet q_f abab \bullet$$

Solución:

$$M = (\{1, a, b, A, B, \bullet\}, \{1, a, b\}, \bullet, \{q_0, q_1 \dots q_{13}\}, q_0, f, \{q_{13}\})$$

Donde la función de transición f , se define de la siguiente manera:



19. (♣) Leonardo Pisano (más conocido por su apodo "Fibonacci") fue un matemático italiano de finales del siglo XII que se hizo famoso por su serie de números:

1 1 2 3 5 8 13 21 ...

Esta serie aparece como solución a un curioso problema:

Si encerramos una pareja de conejos en una granja sin contacto con el exterior, y suponemos que cada mes una pareja de conejos produce una nueva pareja, de forma que a partir del segundo mes la nueva pareja comienza a reproducirse a su vez. ¿Cuántas parejas tendremos al cabo de un mes? ¿Y al cabo de dos? ¿Y al cabo de n meses?

La solución, conocida como serie de Fibonacci, se caracteriza porque cada término es la suma de los dos anteriores:

$$f(n) = f(n-2) + f(n-1)$$

Diseña una máquina de Turing que calcule los n primeros términos de la serie de Fibonacci y **explique su funcionamiento**. Tenga en cuenta que:

- La codificación a utilizar, para todos los números, será una codificación natural (es decir, cada número está representado por tantos unos como su valor natural). Así, la codificación del número 3 será: 111, la del 4: 1111, etcétera.
- En la cinta tenemos como entrada un número n (que indica el número de términos de la serie que se desea hallar), un asterisco y **los dos primeros términos de la serie** separados por un blanco. Nótese, que como se facilitan los dos primeros términos, supondremos que $n \geq 2$.
- En la cinta deben quedar los n primeros términos de la serie de Fibonacci que han sido solicitados.

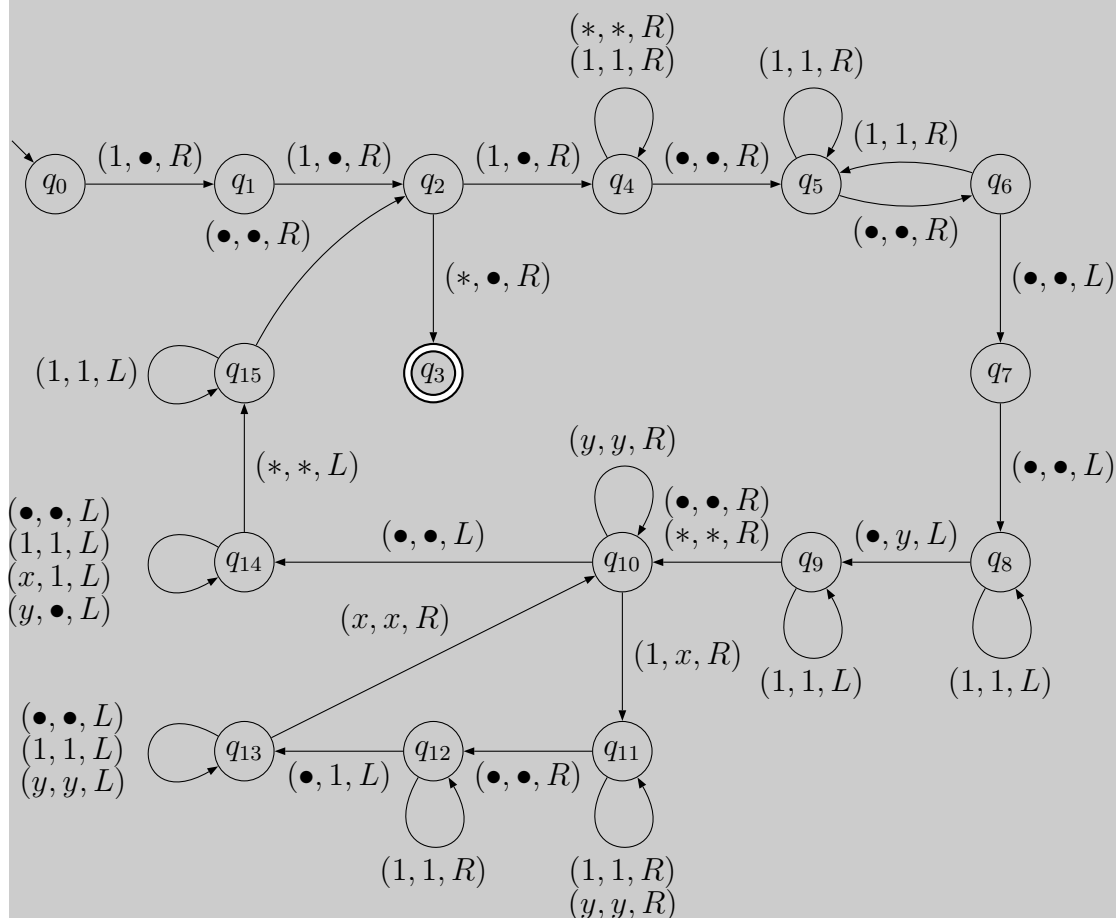
Un ejemplo que genera los tres primeros números de la serie de Fibonacci es:

$$\bullet q_0 111 * 1 \bullet 1 \bullet \vdash^* \bullet q_f 1 \bullet 1 \bullet 11 \bullet$$

Solución:

$$M = (\{1, x, y, *, \bullet\}, \{1, *\}, \bullet, \{q_0, q_1 \dots q_{15}\}, q_0, f, \{q_3\})$$

Donde la función de transición f , se define de la siguiente manera:



20. (♣) En matemáticas, una progresión aritmética es una serie de números tales que la diferencia de dos términos sucesivos cualesquiera de la secuencia es una constante (llamada diferencia de la progresión o simplemente diferencia). Por ejemplo, la sucesión “3, 5, 7, 9, 11,...” es una progresión aritmética cuya diferencia es 2 y cuyo término inicial es el número 3.

Diseña una máquina de Turing que calcule el término n de una progresión aritmética dada y **explica su funcionamiento**.

Se deberán tener en cuenta los siguientes aspectos:

- La codificación a utilizar, para todos los números, será una codificación natural (es decir, cada número estará representado por tantos unos como su valor natural). Así, la codificación del número 3 será: 111, la del 4: 1111, etcétera.
- En la cinta tendremos como entrada un número n que indica el término de la progresión que se desea hallar, un asterisco, la diferencia de la progresión, un blanco y el término inicial. Tanto el término solicitado (n) como la diferencia y el número inicial de la progresión, serán siempre mayores o iguales a 1.
- El término n -ésimo (n) de una progresión de diferencia d y número inicial a , se puede calcular de la siguiente manera:

$$\text{termino}_n = a + (n - 1)d$$

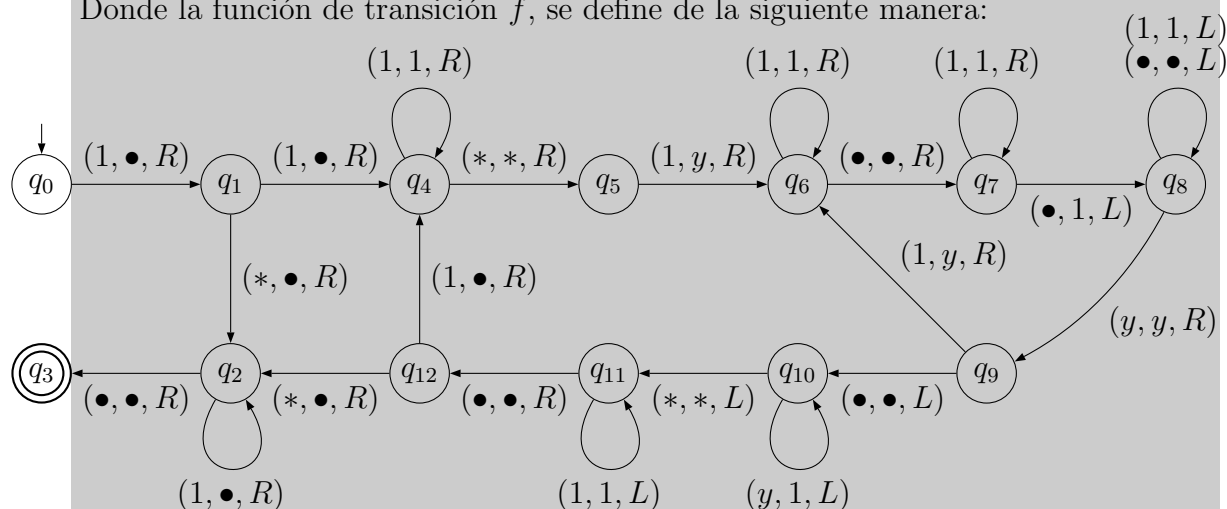
Considérense los siguientes ejemplos:

$$\begin{aligned} \bullet q_0 11 * 1 \bullet 1 \bullet &\vdash^* \bullet q_f 11 \bullet \\ \bullet q_0 11 * 11 \bullet 1 \bullet &\vdash^* \bullet q_f 111 \bullet \\ \bullet q_0 111 * 111 \bullet 1111 \bullet &\vdash^* \bullet q_f 1111111111 \bullet \end{aligned}$$

Solución:

$$M = (\{1, y, *, \bullet\}, \{1, *\}, \bullet, \{q_0, q_1 \dots q_{12}\}, q_0, f, \{q_3\})$$

Donde la función de transición f , se define de la siguiente manera:



El algoritmo se basa en ir calculando de forma incremental el número de la serie que nos solicitan. La máquina de Turing arriba descrita, funciona de la siguiente manera:

- $q_1 \dots q_3$: Como ya hemos calculado el elemento solicitado, eliminamos el asterisco y los unos correspondientes a la diferencia y colocamos el cabezal.
- $q_1 \dots q_9$: Deberemos sumar la diferencia al número de la serie. Para ello, vamos marcando los unos correspondientes a la diferencia.
- $q_9 \dots q_{12}$: Cuando ya hemos sumado la diferencia, desmarcamos los unos y volvemos a ver si tenemos que seguir calculando el siguiente elemento de la serie.

21. (♣) Diseña una máquina de Turing que, dado un número natural, halle su codificación en binario. **Explica su funcionamiento.**

Se deberán tener en cuenta los siguientes aspectos:

- La codificación a utilizar para los números de entrada será una codificación unaria. Así, la codificación del número 3 será: 1111, la del 4: 11111, etcétera.
- En la cinta sólo tendremos como entrada un número n que es el número del que queremos hallar su codificación binaria.

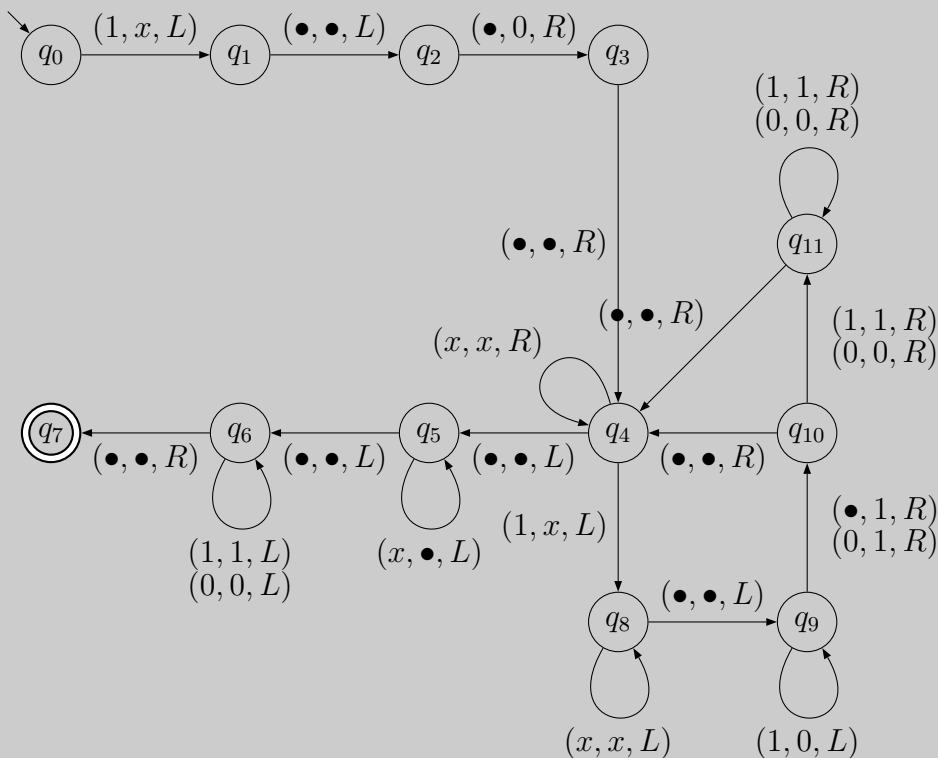
Considérense los siguientes ejemplos:

$$\begin{array}{rcl}
 \bullet q_0 1 \bullet & \vdash^* & \bullet q_f 0 \bullet \\
 \bullet q_0 111 \bullet & \vdash^* & \bullet q_f 10 \bullet \\
 \bullet q_0 1111 \bullet & \vdash^* & \bullet q_f 11 \bullet \\
 \bullet q_0 1111111 \bullet & \vdash^* & \bullet q_f 110 \bullet \\
 \bullet q_0 11111111 \bullet & \vdash^* & \bullet q_f 1000 \bullet
 \end{array}$$

Solución:

$$M = (\{1, x, 0, \bullet\}, \{1\}, \bullet, \{q_0, q_1 \dots q_{11}\}, q_0, f, \{q_7\})$$

Donde la función de transición f , se define de la siguiente manera:



El algoritmo se basa en ir calculando de forma incremental la codificación binaria del número que nos presentan, para ello, el primer uno del número (que está codificado en unario), representa al cero. Una vez colocado el cero en la parte izquierda del número, que es donde vamos a almacenar el resultado, vamos realizando sumas binarias del resultado con 1 por cada elemento que tenga el número que nos dan como entrada. La máquina de Turing arriba descrita, funciona de la siguiente manera:

- $q_1 \dots q_4$: Tratamos el caso especial del número 0.
- $q_4, q_8 \dots q_{11}$: Por cada 1 que nos encontramos en el número, lo marcamos con una “x” y sumamos un 1 al resultado (lo hacemos realizando una suma binaria).
- $q_5 \dots q_7$: Cuando ya hemos tratado todo el número, borramos las “x” que hemos utilizado como marcadores y colocamos el cabezal al inicio del resultado.