

Máquinas de Turing (*Turing Machines*)

Alan Reyes-Figueroa

Teoría de la Computación

(Aula 22) 23.octubre.2023

Definición

Ejemplos

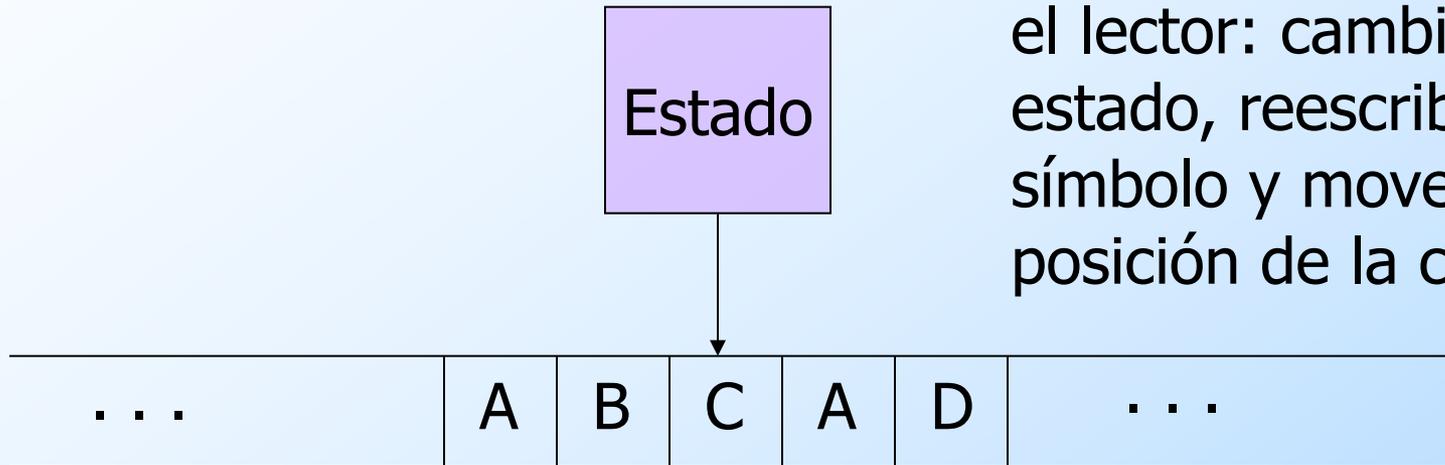
Funciones computables

Máquinas de Turing

- ◆ El propósito de la teoría de las máquinas de Turing es probar que ciertos lenguajes específicos no tienen algoritmo.
- ◆ Las reducciones se utilizan para demostrar que las preguntas más comunes son indecidibles.

Máquinas de Turing

Acción: basada en el estado y el símbolo de la cinta bajo el lector: cambiar de estado, reescribir el símbolo y moverse una posición de la cinta.



Una cinta infinita con espacios (cuadrados) y símbolos elegidos de un alfabeto finito.

¿Por qué Máquinas de Turing?

- ◆ ¿Por qué no trabajar con programas (de C o de Python) o algo parecido?
- ◆ **Respuesta:** Podríamos, pero es más fácil probar propiedades sobre las Máquinas de Turing, porque son simples.
 - ◆ Y sí, son igual de poderosas que cualquier computador.
 - Además, tienen memoria infinita.

¿Por qué no usar Autómatas?

- ◆ En principio, podríamos usarlos, pero no es constructivo.
- ◆ Los modelos de programación no se pueden construir bajo memoria limitada.
 - ◆ Podríamos “comprar más memoria”.
- ◆ Los autómatas finitos son vitales al nivel base (verificación).
Pero no a nivel generalización.

Máquinas de Turing

- ◆ Una Máquina de Turing se describe por:
 1. Un conjunto finito de *estados* (Q).
 2. Un *alfabeto de entrada* (Σ).
 3. Un *alfabeto de cinta* (Γ ; contiene a Σ).
 4. Una *función de transición* (δ).
 5. Un *estado inicial* ($q_0 \in Q$).
 6. Un *símbolo blanco* ($B \in \Gamma - \Sigma$).
 - ◆ Toda la cinta, excepto el input está en blanco.
 7. Un conjunto de *estados finales* ($F \subseteq Q$).

Convenciones

- ◆ a, b, c, \dots son símbolos input.
- ◆ \dots, X, Y, Z son símbolos cinta.
- ◆ \dots, w, x, y, z son cadenas de símbolos input.
- ◆ α, β, \dots son cadenas de símbolos cinta.

Función de Transición

- ◆ $\delta: Q \times \Gamma \rightarrow Q \times \Gamma \times D$ toma dos argumentos:
 1. Un estado, q en Q .
 2. Un símbolo de cinta Z en Γ .
- ◆ $\delta(q, Z)$ está indefinido, o es una tripla de la forma (p, Y, D) , con:
 - ◆ $p \in Q$ un estado,
 - ◆ $Y \in \Gamma$ un símbolo de cinta,
 - ◆ $D \in \{L, R\}$ una *dirección*, (*Left* ó *Right*).

Acciones de una MT

- ◆ Si $\delta(q, Z) = (p, Y, D)$ entonces, en el estado q , leyendo el símbolo Z (en el lector de la cinta), la máquina de Turing hace lo siguiente:
 1. Cambia al estado p .
 2. Reemplaza Z por Y en la cinta.
 3. Mueve una posición el lector de la cinta, en la dirección D .
 - ◆ $D = L$: se mueve a la izquierda; $D = R$; a la derecha.

Ejemplo: Máquina de Turing

- ◆ Construimos una máquina de Turing que lee el input hacia la derecha, buscando un 1.
- ◆ Si encuentra uno, lo modifica a 0, va al estado final f , y termina.
- ◆ Si alcanza un símbolo blanco, lo cambia a 1 y se mueve hacia la izquierda.

Ejemplo: Máquina de Turing

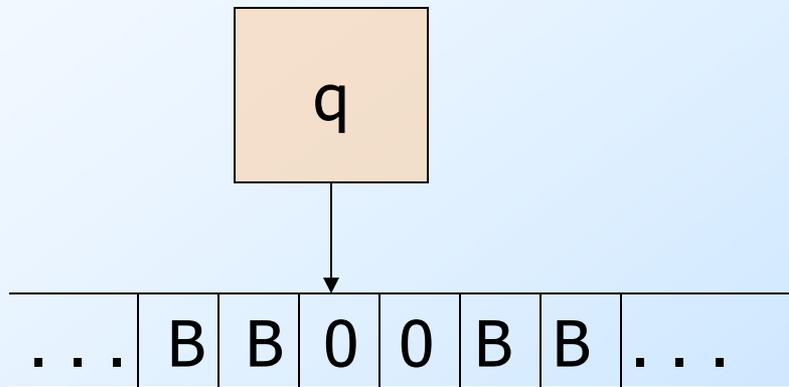
- ◆ Estados: $Q = \{q \text{ (inicial)}, f \text{ (final)}\}$.
- ◆ Símbolos input: $\Sigma = \{0, 1\}$.
- ◆ Símbolos cinta: $\Gamma = \{0, 1, B\}$. ($B = _$)
- ◆ Función de Transición:
 - ▶ $\delta(q, 0) = (q, 0, R)$. $\delta(f, 0) = (f, 0, R)$
 - ▶ $\delta(q, 1) = (f, 0, R)$. $\delta(f, 1) = (f, 1, R)$.
 - ▶ $\delta(q, B) = (q, 1, L)$. $\delta(f, B) = (f, B, \text{halt})$.

Simulación

$$\delta(q, 0) = (q, 0, R)$$

$$\delta(q, 1) = (f, 0, R)$$

$$\delta(q, B) = (q, 1, L)$$

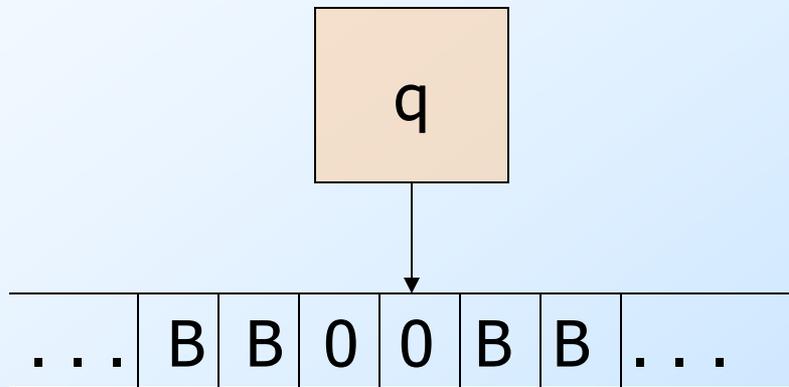


Simulación

$$\delta(q, 0) = (q, 0, R)$$

$$\delta(q, 1) = (f, 0, R)$$

$$\delta(q, B) = (q, 1, L)$$

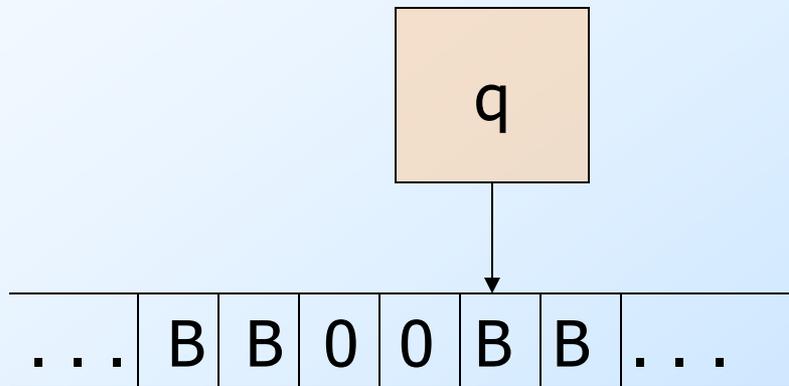


Simulación

$$\delta(q, 0) = (q, 0, R)$$

$$\delta(q, 1) = (f, 0, R)$$

$$\delta(q, B) = (q, 1, L)$$

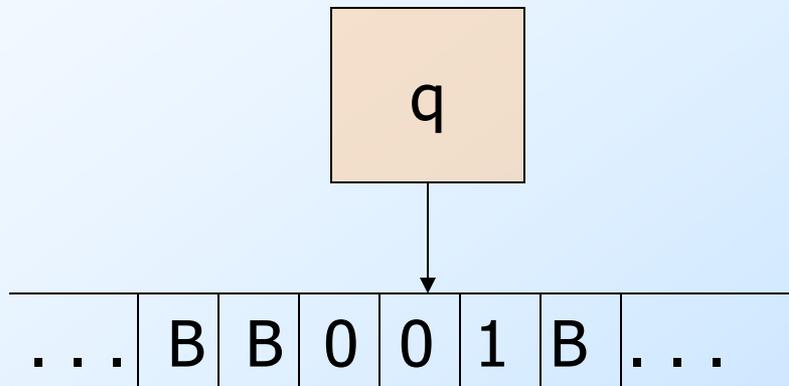


Simulación

$$\delta(q, 0) = (q, 0, R)$$

$$\delta(q, 1) = (f, 0, R)$$

$$\delta(q, B) = (q, 1, L)$$

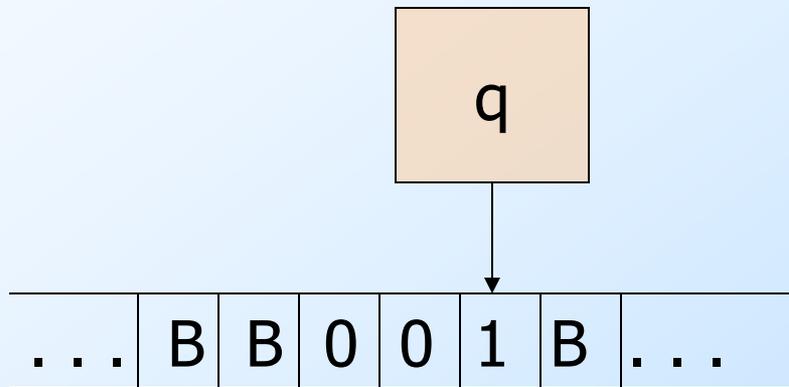


Simulación

$$\delta(q, 0) = (q, 0, R)$$

$$\delta(q, 1) = (f, 0, R)$$

$$\delta(q, B) = (q, 1, L)$$

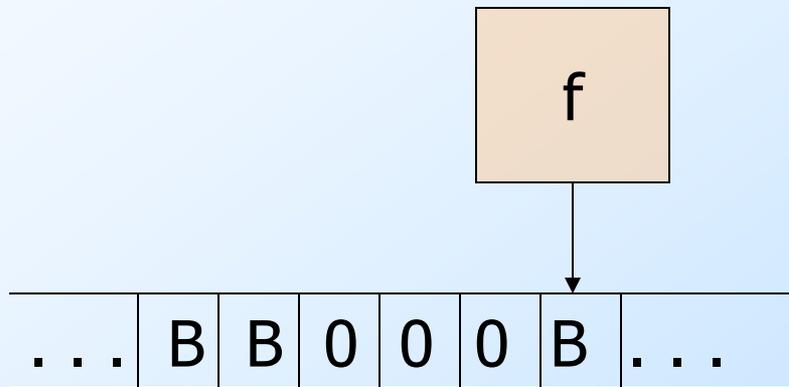


Simulación

$$\delta(q, 0) = (q, 0, R)$$

$$\delta(q, 1) = (f, 0, R)$$

$$\delta(q, B) = (q, 1, L)$$

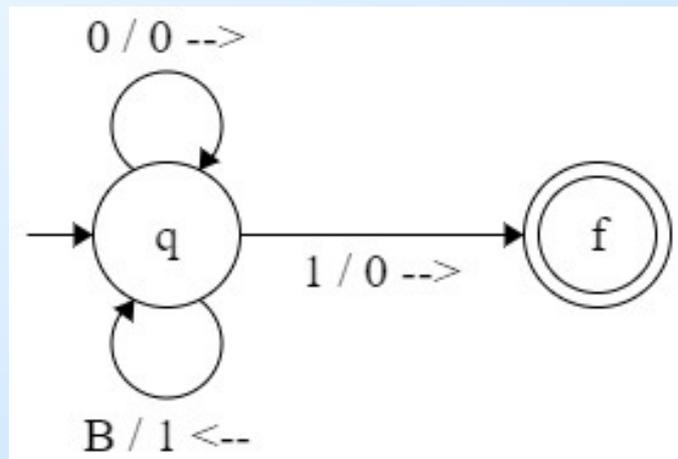


No hay más
movidas posibles.
La máquina para
y acepta.

Tabla de Transiciones

	Símbolo		
Estado	0	1	B
q	(q, 0, R)	(f, 0, R)	(q, 1, L)
f	-	-	-

Diagrama de Transiciones



Descripciones Instantáneas

- ◆ Como la cinta es infinita, se representan solo los símbolos entre los B's (a veces se pueden incluir algunos B's) y
- ◆ se incluye un símbolo especial para indicar la posición del lector cinta.
- ◆ Por ejemplo:

$$\alpha q \beta = X_1 X_2 \dots X_{i-1} q X_i X_{i+1} \dots X_n$$

representa una descripción instantánea donde:

- ◆ q es el estado de la máquina de Turing
- ◆ la cabeza de la cinta está viendo al i -ésimo símbolo a la izquierda
- ◆ $X_1 X_2 \dots X_n$ es el pedazo de cinta entre los símbolos más a la izquierda y más a la derecha que no son vacíos.

Descripciones Instantáneas

- ◆ Usamos la misma notación de descripción instantánea que en los autómatas de pila: \vdash y \vdash^* .
 - ◆ \vdash "se convierte en un movimiento",
 - ◆ \vdash^* "se convierte en cero o más movim."
- ◆ **Ejemplo:** Los movimientos de la MT anterior son
 $q_00 \vdash 0q_0 \vdash 00q_1 \vdash 0q_01 \vdash 00q_1 \vdash 000f$

Definición Formal de \vdash

1. Si $\delta(q, Z) = (p, Y, R)$, entonces escribimos:

$$\alpha q Z \beta \vdash \alpha Y p \beta$$

Si Z es el símbolo blanco B , entonces

$$\alpha q \vdash \alpha Y p$$

1. Si $\delta(q, Z) = (p, Y, L)$, entonces escribimos:

► Para cualquier X , $\alpha X q Z \beta \vdash \alpha p X Y \beta$

► Además, $q Z \beta \vdash p B Y \beta$

Ejemplo:

- ◆ Las descripciones instantáneas para el ejemplo anterior son:

$\delta(q, 0) = (q, 0, R)$	q00	⊢	0q0
$\delta(q, 0) = (q, 0, R)$		⊢	00q
$\delta(q, B) = (q, 1, L)$		⊢	0q01
$\delta(q, 0) = (q, 0, R)$		⊢	00q1
$\delta(q, 1) = (f, 0, R)$		⊢	000f
		⊢	000

- ◆ Así, $q000 \vdash^* 000$.

Ejemplo 2:

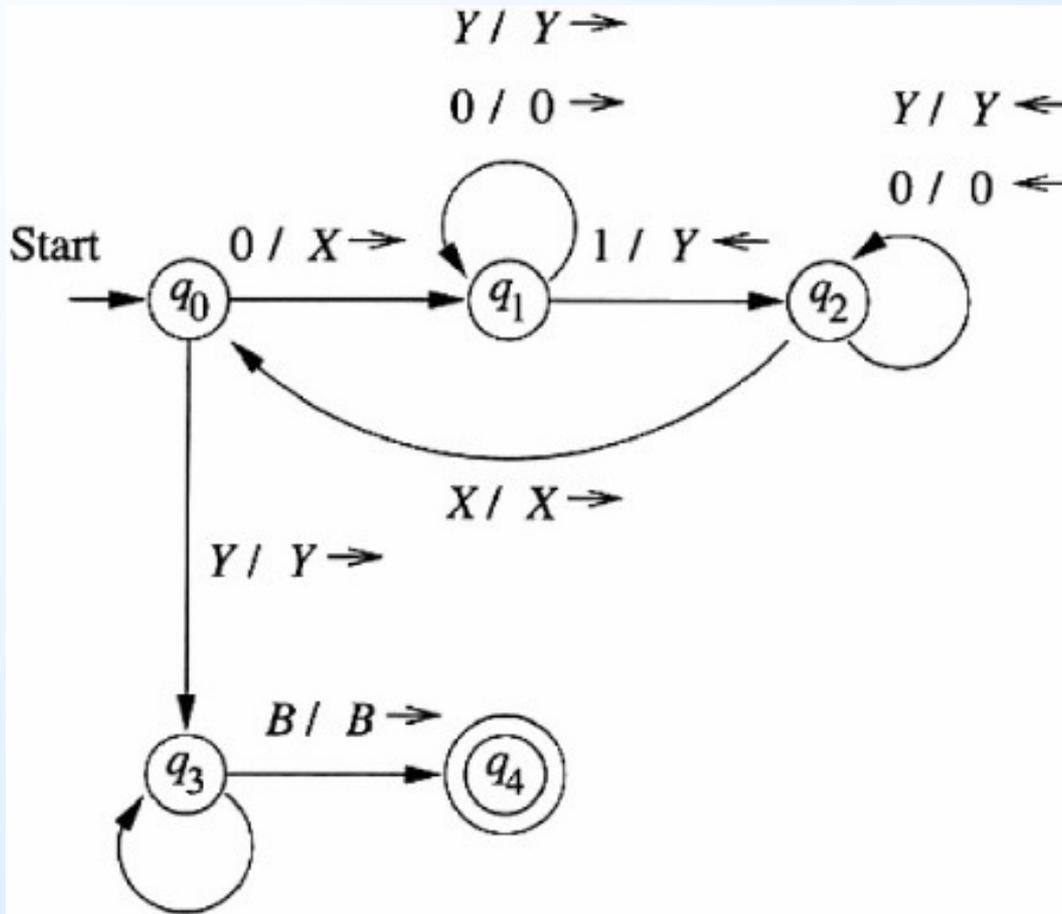
- ◆ Construimos una máquina de Turing que acepta las cadenas $\{0^n 1^n: n \geq 1\}$.
- ◆ Estados: $Q = \{q_0, q_1, q_2, q_3, q_4\}$.
- ◆ Símbolos input: $\Sigma = \{0, 1\}$.
- ◆ Símbolos cinta: $\Gamma = \{0, 1, X, Y, B\}$.
- ◆ Estado inicial: q_0
- ◆ Estados de aceptación: $F = \{q_4\}$.

Ejemplo 2:

► Función de Transición:

	Símbolo				
Estado	0	1	X	Y	B
q0	(q1, X, R)	-	-	(q3, Y, R)	-
q1	(q1, 0, R)	(q2, Y, L)	-	(q1, Y, R)	-
q2	(q2, 0, L)	-	(q0, X, R)	(q2, Y, L)	-
q3	-	-	-	(q3, Y, R)	(q4, B, R)
q4	-	-	-	-	-

Ejemplo 2:



Ejercicio: $w = 0011$.

Ejemplo 2:

- ◆ Para la cadena $w = 0011$, las transiciones son:

$\delta(q_0, 0) = (q_1, X, R)$	q_00011	⊢	Xq_1011
$\delta(q_1, 0) = (q_1, 0, R)$		⊢	$X0q_111$
$\delta(q_1, 1) = (q_2, 1, L)$		⊢	Xq_20Y1
$\delta(q_2, 0) = (q_2, X, L)$		⊢	q_2X0Y1
$\delta(q_2, X) = (q_0, X, R)$		⊢	Xq_00Y1
$\delta(q_0, 0) = (q_1, X, R)$		⊢	XXq_1Y1
$\delta(q_1, Y) = (q_1, Y, R)$		⊢	$XXYq_11$
$\delta(q_1, 1) = (q_2, 1, L)$		⊢	XXq_2YY
$\delta(q_2, Y) = (q_2, Y, L)$		⊢	XXq_2YY
$\delta(q_2, X) = (q_0, X, R)$		⊢	Xq_2XYY
$\delta(q_0, Y) = (q_3, Y, R)$		⊢	XXq_0YY
$\delta(q_0, Y) = (q_3, Y, R)$		⊢	$XXYq_3Y$
$\delta(q_0, Y) = (q_3, Y, R)$		⊢	$XXYYq_3B$
$\delta(q_3, B) = (q_4, B, R)$		⊢	$XXYYBq_4$

Lenguajes de una MT

- ◆ Una máquina de Turing define un lenguaje por estados de aceptación.
 - ▶ $L(M) = \{w: q_0w \vdash^* I, \text{ con } I \text{ una descripción instantánea con estado final}\}$.
- ◆ Alternativamente, una máquina de Turing define un lenguaje mediante paro.
 - ▶ $H(M) = \{w: q_0w \vdash^* I, \text{ si ya no hay movidas posibles desde la descripción } I\}$.

Equivalencia de Paro y Aceptación

1. Si $L = L(M)$, entonces existe una máquina de Turing M' tal que
$$L = H(M').$$
2. Si $L = H(M)$, entonces existe una máquina de Turing M'' tal que
$$L = L(M'').$$

Prueba de (1): Aceptación \rightarrow Paro

- ◆ Modificar M para definir M' como sigue:
 1. Para cada estado de aceptación de M , remover cualquier transición, así M' para en ese estado.
 2. Evitar que M' pare de forma accidental.
 - ◆ Introducir un nuevo estado s , que corre a la derecha infinitamente; esto es $\delta(s, X) = (s, X, R)$ para todo símbolo X .
 - ◆ Si q no es estado de aceptación, y $\delta(q, X)$ es indefinida, entonces $\delta(q, X) = (s, X, R)$.

Prueba de (2): Paro \rightarrow Aceptación

- ◆ Modificar M para definir M'' como sigue:
 1. Introducir un nuevo estado f , el único estado de aceptación de M'' .
 2. f no posee transiciones.
 3. Si $\delta(q, X)$ es indefinida para cualquier estado q y símbolo X , la definimos como $\delta(q, X) = (f, X, R)$.

Lenguajes Recursivamente Enumerables

- ◆ Hemos visto que las clases de lenguajes definidos por máquinas de Turing que usan estado final o paro son las mismas.
- ◆ Esta clase de lenguajes se denomina *lenguajes recursivamente enumerables*.
- ◆ El término en realidad es anterior a las máquinas de Turing y se refiere a otra noción de cálculo de funciones.

Lenguajes Recursivamente Enumerables

- ◆ Un *algoritmo* es una máquina de Turing M para la que está garantizado que para, ya sea que acepte o no.
- ◆ Si $L = L(M)$ para alguna máquina de Turing M que es un algoritmo, decimos que L es un *lenguaje recursivo*.

Ejemplo: Lenguajes Recursivos

- ◆ Todo lenguaje libre de contexto CFL es un lenguaje recursivo.
 - ▶ Usar el algoritmo CYK.
- ◆ Todo lenguaje regular es un CFL (piense en su DFA como un autómatas de pila que ignora su pila); por lo tanto, todo lenguaje regular es recursivo.
- ◆ “Casi todo” lo que uno cree es recursivo.

Ejemplo:

- ◆ Usamos el PDA del ejemplo anterior. Tenemos:

$$\begin{array}{l} (q, 000111, Z_0) \vdash (q, 00111, XZ_0) \\ \vdash (q, 0111, XXZ_0) \\ \vdash (q, 111, XXXZ_0) \\ \vdash (p, 11, XXZ_0) \\ \vdash (p, 1, XZ_0) \\ \vdash (p, \epsilon, Z_0) \\ \vdash (f, \epsilon, Z_0) \end{array}$$

- ◆ Así, $(q, 000111, Z_0) \vdash^* (f, \epsilon, Z_0)$.
- ◆ Ejercicio: ¿Cómo quedaría si el input fuese 0001111?