Máquinas de Turing

José de Jesús Lavalle Martínez

Benemérita Universidad Autónoma de Puebla
Facultad de Ciencias de la Computación
Lenguajes Formales y Autómatas CCOS 014

Contenido

Un lenguaje que no es libre de contexto

Ejemplo 1

Considere el siguiente lenguaje:

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

Tres intentos de solución:

Un lenguaje que no es libre de contexto

Ejemplo 1

Considere el siguiente lenguaje:

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

Tres intentos de solución:

1.
   1. Empilar las as,
   2. Por cada b que leamos se desempila una a, pero no hay contra quien comparar las cs porque en la pila sólo quedaría el símbolo de pila inicial (suponiendo que se leyeron el mismo número de as que de bs).

Un lenguaje que no es libre de contexto

Ejemplo 1

Considere el siguiente lenguaje:

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

Tres intentos de solución:

2.
   1. Empilar las as y bs,
   2. Por cada c que leamos se desempila una b, pero no hay contra quien comparar las as que quedan en la pila, ya que las bs ya se desempilaron y las cs ya se leyeron (suponiendo que se leyeron el mismo número de bs que de cs).

Ejemplo 1

Considere el siguiente lenguaje:

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

Tres intentos de solución:

3.
   1. Empilar las as, bs y cs,
   2. Como ya terminamos de leer la cadena de entrada, no tenemos contra quien comparar los símbolos que están en la pila.

Descripción Máquina de Turing

- Una máquina de Turing es un autómata cuyo almacenamiento temporal es una cinta infinita en ambas direcciones.

- La cinta se divide en celdas, cada una de éstas puede almacenar un solo símbolo. Como la cinta es infinita se asume que todas las celdas están en blanco, excepto una secuencia finita no vacía de celdas contiguas que no están en blanco, estas celdas al inicio contienen la entrada de la máquina de Turing y cuando la máquina pare contienen la salida de la máquina de Turing.

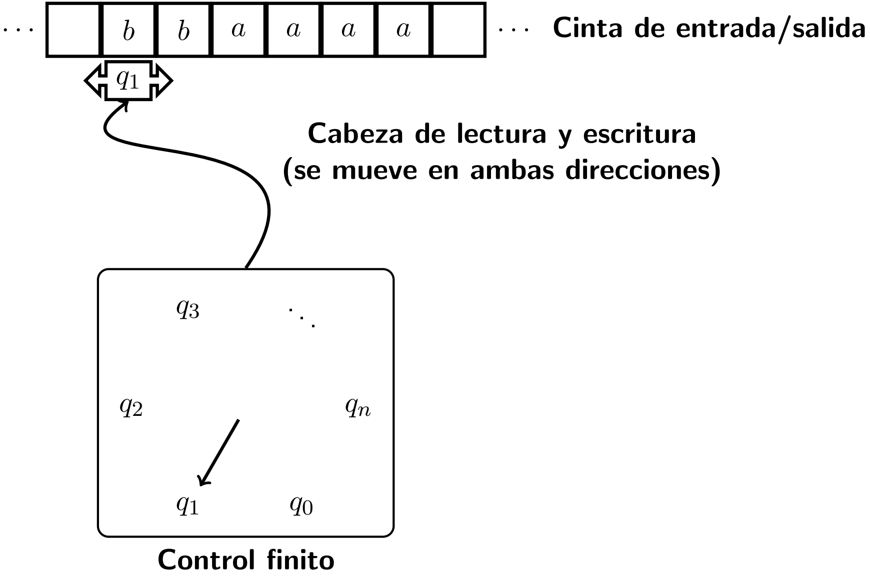
Descripción Máquina de Turing

- Asociada a la cinta, tiene una cabeza de lectura-escritura que puede viajar a la derecha e izquierda de la cinta y puede leer y escribir un solo símbolo en cada movimiento.

Descripción Máquina de Turing

- Como en todos los autómatas, tiene un control finito constituido por los estados internos y la función de transición.

Diagrama Máquina de Turing



Definición 2

Una máquina de Turing se define por la septupla

$$M = (Q, \Sigma, \Gamma, \delta, q_0, \Box, F),$$

donde

Q es el conjunto de estados internos de la unidad de control,

Σ es el alfabeto de entrada,

Γ es un conjunto finito de de símbolos llamado el **alfabeto de la cinta**,

$\delta : Q \times \Gamma \to Q \times \Gamma \times \{L, R\}$ es la función de transición,

$q_0 \in Q$ es el estado inicial de la unidad de control,

$\Box \in \Gamma$ es un símbolo especial llamado **blanco**,

$F \subseteq Q$ es el conjunto de estados finales.

Se asume que $\Sigma \subseteq \Gamma - \{\Box\}$.

La función de transición

La función de transición formaliza la manera en que opera la máquina de Turing:

$$\delta : Q \times \Gamma \to Q \times \Gamma \times \{L, R\}$$

1. Asumiremos que es una función parcial aunque determinista, en el sentido que a lo más hay una entrada para cada par (q, a) del dominio.

La función de transición

La función de transición formaliza la manera en que opera la máquina de Turing:

$$\delta : Q \times \Gamma \to Q \times \Gamma \times \{L, R\}$$

2. Sus argumentos son el estado actual de la unidad de control y el símbolo de cinta que esta leyendo actualmente.

La función de transición

La función de transición formaliza la manera en que opera la máquina de Turing:

$$\delta : Q \times \Gamma \to Q \times \Gamma \times \{L, R\}$$

3. El resultado es un nuevo estado de la unidad de control, un símbolo de cinta nuevo que reemplaza al que leyó y un símbolo de movimiento L o R.

La función de transición

La función de transición formaliza la manera en que opera la máquina de Turing:

$$\delta : Q \times \Gamma \to Q \times \Gamma \times \{L, R\}$$

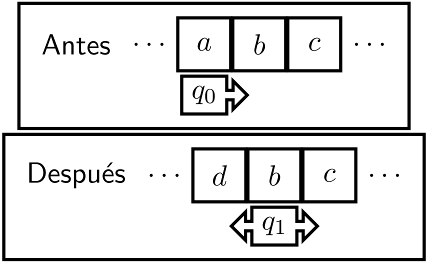
4. El símbolo de movimiento indica si la cabeza de lectura-escritura se mueve una celda a la izquierda o la derecha, después de que se haya escrito en la cinta el símbolo nuevo.

Ejemplo como opera la función de transición I

Considere el siguiente caso de la función de transición:

$$\delta(q_0, a) = (q_1, d, R)$$

El antes y después de realizar el movimiento se muestra a continuación:

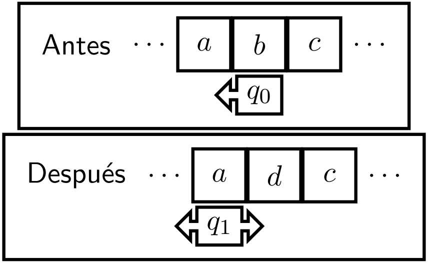


Ejemplo como opera la función de transición II

Considere el siguiente caso de la función de transición:

$$\delta(q_0, b) = (q_1, d, L)$$

El antes y después de realizar el movimiento se muestra a continuación:



Funcionamiento de la máquina de Turing

1. La máquina empieza en el estado inicial con alguna cadena sobre la cinta.

Funcionamiento de la máquina de Turing

2. Luego realiza una secuencia de pasos controlados por la función de transición δ.

3. Durante este proceso, el contenido de cualquier celda puede ser examinado y cambiando muchas veces.

Funcionamiento de la máquina de Turing

4. Eventualmente este proceso puede terminar, en cuyo caso se dice que la máquina de Turing alcanza un **estado de paro**.

5. Se dice que una máquina de Turing **para** si alcanza una configuración para la cual no está definida δ, lo cual es posible porque asumimos que δ es una función parcial.

Funcionamiento de la máquina de Turing

6. También asumiremos que para los estados finales no se definirán transiciones, así la máquina de Turing parará siempre que entre a un estado final.

Ejemplo I

Ejemplo 3

Considere la máquina de Turing definida mediante:

$$
\begin{aligned}
Q &= \{q_0, q_1\}, & \delta(q_0, a) &= (q_0, b, R),\\
\Sigma &= \{a, b\}, & \delta(q_0, b) &= (q_0, b, R),\\
\Gamma &= \{a, b, \square\}, & \delta(q_0, \square) &= (q_1, \square, L).\\
F &= \{q_1\}, & &
\end{aligned}
$$

1. Inicia en el estado q_0, si lee una a la cambia por b, si lee una b la reescribe, en ambos casos se mantine en q_0 y se mueve a la derecha.

Ejemplo 3

Considere la máquina de Turing definida mediante:

$$Q = \{q_0, q_1\}, \qquad \delta(q_0, a) = (q_0, b, R),$$
$$\Sigma = \{a, b\}, \qquad \delta(q_0, b) = (q_0, b, R),$$
$$\Gamma = \{a, b, \square\}, \qquad \delta(q_0, \square) = (q_1, \square, L).$$
$$F = \{q_1\},$$

2. Cualquier a que encuentre en su camino hacia la derecha será cambiada por b, las bs se reescribirán.

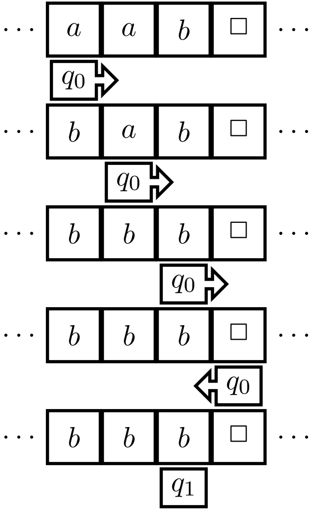
Ejemplo 3

Considere la máquina de Turing definida mediante:

$$
\begin{aligned}
Q &= \{q_0, q_1\}, & \delta(q_0, a) &= (q_0, b, R),\\
\Sigma &= \{a, b\}, & \delta(q_0, b) &= (q_0, b, R),\\
\Gamma &= \{a, b, \Box\}, & \delta(q_0, \Box) &= (q_1, \Box, L).\\
F &= \{q_1\}, & &
\end{aligned}
$$

3. Cuando la máquina encuentre el primer blanco, cambia al estado q_1, reescribe el blanco y se mueve a la izquierda, como q_1 es un estado de paro la máquina de Turing para, como $q_1 \in F$, la máquina de Turing para en un estado final.

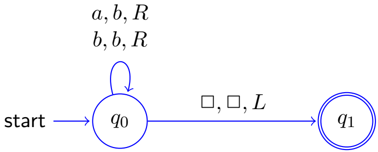
Ejemplo II



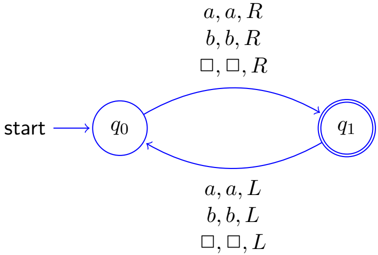
Ejemplo III

Como es de esperarse, también podemos usar grafos de transición para representar máquinas de Turing.

$$
\begin{aligned}
Q &= \{q_0, q_1\}, & \delta(q_0, a) &= (q_0, b, R),\\
\Sigma &= \{a, b\}, & \delta(q_0, b) &= (q_0, b, R),\\
\Gamma &= \{a, b, \square\}, & \delta(q_0, \square) &= (q_1, \square, L).\\
F &= \{q_1\}, & &
\end{aligned}
$$



Otro ejemplo



1. $\square\overset{q_0}{\square}\square \rightarrow \square\square\overset{q_1}{\square} \rightarrow \square\overset{q_0}{\square}\square \rightarrow \cdots$
2. $\square\overset{q_0}{a}\square \rightarrow \square a\overset{q_1}{\square} \rightarrow \square\overset{q_0}{a}\square \rightarrow \cdots$
3. $\square\overset{q_0}{a}b\square \rightarrow \square a\overset{q_1}{b}\square \rightarrow \square\overset{q_0}{a}b\square \rightarrow \cdots$
4. $\square\overset{q_0}{a}ba\square \rightarrow \square a\overset{q_1}{b}a\square \rightarrow \square\overset{q_0}{a}ba\square \rightarrow \cdots$

Máquina de Turing estándar

Hay diferentes definiciones de Máquinas de Turing, la definición que utilizamos la llamaremos **máquina de Turing estándar** y tiene las siguientes características:

1. La máquina de Turing tiene una cinta que no está acotada ni a la izquierda ni a la derecha, por lo tanto permite cualquier número de movimientos hacia la izquierda y hacia la derecha

Hay diferentes definiciones de Máquinas de Turing, la definición que utilizamos la llamaremos **máquina de Turing estándar** y tiene las siguientes características:

2. La máquina de Turing es determinista en el sentido que δ define a lo más un movimiento para cada configuración posible.

Hay diferentes definiciones de Máquinas de Turing, la definición que utilizamos la llamaremos **máquina de Turing estándar** y tiene las siguientes características:

3. No existe un archivo de entrada especial. Se asume que al inicio la cinta tiene un contenido específico finito (no blanco) y al menos una parte de este contenido es la entrada.

Máquina de Turing estándar

Hay diferentes definiciones de Máquinas de Turing, la definición que utilizamos la llamaremos **máquina de Turing estándar** y tiene las siguientes características:

4. Similarmente, no existe un dispositivo de salida. Siempre que la máquina para, una parte o todo el contenido (no blanco) de la cinta se considera como la salida.

Descripción instantánea I

Como en el caso de APND, usaremos el concepto de descripcion instantánea. Note que cualquier configuración de la máquina de Turing está completamente determinada por:

- El estado actual de la unidad de control,
- el contenido de la cinta y
- la posición de la cabeza de lectura-escritura.

Definición 4

Supongamos que el contenido de la cinta de una máquina de Turing es:

$$a_1 a_2 \cdots a_{k-1} a_k a_{k+1} \cdots a_n,$$

que

$$x_1 = a_1 a_2 \cdots a_{k-1} \text{ y } x_2 = a_k a_{k+1} \cdots a_n,$$

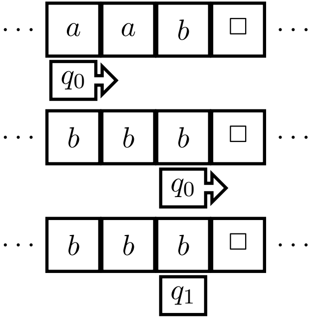
y que estando en el estado q la cabeza de lectura-escritura está posicionada en el símbolo a_k, entonces,

$$x_1 q x_2$$

o

$$a_1 a_2 \cdots a_{k-1} q a_k a_{k+1} \cdots a_n$$

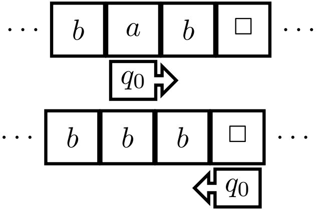
es la **descripción instantánea** de la máquina de Turing.



q_0aab

bbq_0b

bbq_1b



bq_0ab

$bbbq_0\Box$

Movimiento en un paso

Definición 5

El **movimiento en un paso**, denotado por $\vdash$, de una configuración a otra se define mediante los dos casos siguientes:

1.
$$a_1a_2\cdots a_{k-1}qa_ka_{k+1}\cdots a_n \vdash a_1a_2\cdots a_{k-1}a'_kq'a_{k+1}\cdots a_n$$
si y sólo si
$$\delta(q, a_k) = (q', a'_k, R)$$

2.
$$a_1a_2\cdots a_{k-1}qa_ka_{k+1}\cdots a_n \vdash a_1a_2\cdots q'a_{k-1}a'_ka_{k+1}\cdots a_n$$
si y sólo si
$$\delta(q, a_k) = (q', a'_k, L)$$

La cerradura reflexiva y transitiva de $\vdash$, en símbolos $\overset{*}{\vdash}$, denota el **movimiento en un número arbitrario de pasos**.

Definición 6

Se dice que una máquina de Turing M **para** iniciando en alguna configuración inicial $x_1 q_i x_2$ si

$$x_1 q_i x_2 \overset{*}{\vdash} y_1 q_j a y_2$$

para cualquier q_j y a para los que $\delta(q_j, a)$ está indefinida. A la secuencia de configuraciones que llevan a la máquina M a un estado de paro se le llama **cálculo**.

Ejemplo de movimiento

Ejemplo 7

Considere la máquina de Turing definida mediante:

$$
\begin{aligned}
Q &= \{q_0, q_1\}, & \delta(q_0, a) &= (q_0, b, R),\\
\Sigma &= \{a, b\}, & \delta(q_0, b) &= (q_0, b, R),\\
\Gamma &= \{a, b, \Box\}, & \delta(q_0, \Box) &= (q_1, \Box, L).\\
F &= \{q_1\}, & &
\end{aligned}
$$

Los movimientos de esta máquina de Turing sobre la palabra aab se pueden representar mediante:

$$q_0aab \vdash bq_0ab \vdash bbq_0b \vdash bbbq_0\Box \vdash bbq_1b$$

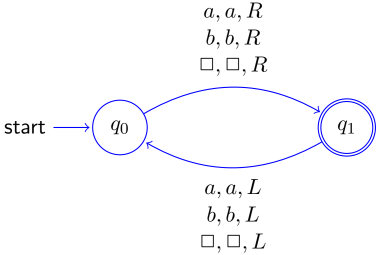
o

$$q_0aab \overset{*}{\vdash} bbq_1b.$$

Ya que $\delta(q_1, b)$ no está definida se dice que la máquina para.

Notación no para

Vimos que la máquina de Turing representada por el grafo de dependencias



nunca para, entra en un ciclo infinito del cual no puede escapar. Esta situación juega un papel fundamental en la discusión de máquinas de Turing y la representamos mediante:

$$x_1 q x_2 \overset{*}{\vdash} \infty$$

indicando que, iniciando en la configuración inicial $x_1 q x_2$, la máquina entra en un ciclo infinito y nunca para.

Definición 8

Sea $M = (Q, \Sigma, \Gamma, \delta, q_0, \Box, F)$ una máquina de Turing. Entonces el lenguaje aceptado por M es

$$L(M) = \left\{ w \in \Sigma^+ : q_0 w \overset{*}{\vdash} x_1 q_f x_2, q_f \in F, x_1, x_2 \in \Gamma^* \right\}.$$

Note que por definición las máquinas de Turing no aceptan la cadena vacía, siempre asumiremos que, cuando la máquina empieza a operar, en la cinta hay una secuencia finita no vacía de símbolos contiguos w y que en el resto de la cinta las celdas contienen $\Box$.

Ejemplo 9

Para $\Sigma = \{0, 1\}$ diseñe una máquina de Turing que acepte el lenguaje denotado por la expresión regular 00^*.

Solución:

1. Empezamos en q_0 leyendo el primer símbolo de la palabra de entrada, si es 0 lo reemplazamos por 0 y continuamos a la derecha sin cambiar de estado.
2. Si la cadena sólo tiene 0s alcanzaremos el primer $\square$ a la derecha de la cadena, en cuyo caso cambiamos al estado final q_1, el símbolo lo reemplazamos por $\square$ y nos movemos a la derecha (o a la izquierda, en este caso es indistinto).

Así tenemos $Q = \{q_0, q_1\}$, $F = \{q_1\}$ y

$$\delta(q_0, 0) = (q_0, 0, R),$$
$$\delta(q_0, \Box) = (q_1, \Box, R).$$

Note que:

- $\delta(q_0, 1)$ no está definida, por lo tanto si la cadena de entrada contiene un 1, entonces la máquina para en un estado que no es final, por lo tanto no acepta la cadena.
- Note que la máquina de Turing pararía en un estado final si empieza leyendo un $\Box$, pero este caso está excluido por la Definición 8.

Ejemplo 10

Para $\Sigma = \{a, b\}$ diseñe una máquina de Turing que acepte

$$L = \{a^n b^n : n \geq 1\}.$$

Ejemplo 10

Para $\Sigma = \{a, b\}$ diseñe una máquina de Turing que acepte

$$L = \{a^n b^n : n \geq 1\}.$$

Solución: Recomendaciones generales:

Ejemplo 10

Para $\Sigma = \{a, b\}$ diseñe una máquina de Turing que acepte

$$L = \{a^n b^n : n \geq 1\}.$$

Solución: Recomendaciones generales:

- Recordemos que podemos recorrer la cinta de izquierda a derecha y de derecha a izquierda un número arbitrario de veces.

Ejemplo 10

Para $\Sigma = \{a, b\}$ diseñe una máquina de Turing que acepte

$$L = \{a^n b^n : n \geq 1\}.$$

Solución: Recomendaciones generales:

- Recordemos que la máquina empieza a operar estando en el estado q_0 y la cabeza de lectura-escritura está leyendo el primer símbolo de la cadena de entrada.

Ejemplo 10

Para $\Sigma = \{a, b\}$ diseñe una máquina de Turing que acepte

$$L = \{a^n b^n : n \geq 1\}.$$

Solución: Recomendaciones generales:

- Es buena estrategia pensar que cada estado debe hacer una tarea específica pero invariante, así diseñaremos máquinas de Turing legibles y eficientes.

Ejemplo 10

Para $\Sigma = \{a, b\}$ diseñe una máquina de Turing que acepte

$$L = \{a^n b^n : n \geq 1\}.$$

Solución: Recomendaciones generales:

- Base su diseño pensando sólo en las cadenas que debe aceptar, olvídese de las que no debe aceptar, si está bien el diseño, quedarán implícitamente descartadas, porque no tienen la forma de las que sí se deben aceptar.

Ejemplo 2 MTAL II

La configuración inicial está dada por:

$$q_0 \overbrace{aa \cdots a}^{n} \overbrace{bb \cdots b}^{n},$$

Ejemplo 2 MTAL II

La configuración inicial está dada por:

$$q_0 \overbrace{aa\cdots a}^{n}\overbrace{bb\cdots b}^{n},$$

1. La idea general es que por cada a, debemos ir hacía la derecha a buscar su correspondiente b y luego regresar para hacer lo mismo con la siguiente a y su correspondiente b.

Ejemplo 2 MTAL II

La configuración inicial está dada por:

$$q_0 \overbrace{aa\cdots a}^{n}\overbrace{bb\cdots b}^{n},$$

1. La idea general es que por cada a, debemos ir hacía la derecha a buscar su correspondiente b y luego regresar para hacer lo mismo con la siguiente a y su correspondiente b.
2. Pero debemos marcar de alguna manera las as y bs que vamos empatando, lo podemos hacer, por ejemplo, cambiando cada a por x y cada b por y.

Ejemplo 2 MTAL II

La configuración inicial está dada por:

$$q_0 \overbrace{aa\cdots a}^{n}\overbrace{bb\cdots b}^{n},$$

1. La idea general es que por cada a, debemos ir hacía la derecha a buscar su correspondiente b y luego regresar para hacer lo mismo con la siguiente a y su correspondiente b.
2. Pero debemos marcar de alguna manera las as y bs que vamos empatando, lo podemos hacer, por ejemplo, cambiando cada a por x y cada b por y.

Así la configuración final debe ser:

$$\overbrace{xx\cdots x}^{n}\overbrace{yy\cdots y}^{n-1} q_f y,$$

Ejemplo 2 MTAL III

Ahora asignemos tareas a cada estado, recuerde que de antemano no sabemos cuántos estados necesitaremos.

1. Estando en q_0 y leyendo la a más izquierda, la máquina cambia al estado q_1 cambia a por x y se mueve a la derecha, es decir:
$$\delta(q_0, a) = (q_1, x, R).$$
2. Estando en q_1 la máquina respeta las as que encuentre en su camino hacia la derecha hasta llegar a la primera b, en cuyo caso la cambia por y, pasa al estado q_2 y se mueve a la izquierda. Note que después de marcar la primera b, q_1 debe respetar también las ys que encuentre en su camino hasta llegar a la b más izquierda, así:
$$\begin{aligned} \delta(q_1, a) &= (q_1, a, R), \\ \delta(q_1, y) &= (q_1, y, R), \\ \delta(q_1, b) &= (q_2, y, L), \end{aligned}$$

3. Estando en q_2 la máquina va hacia la izquierda, en su camino va a encontrar ys y as, las cuales debe respetar, cuando encuentre la última x procesada la respeta y vuelve al inicio del proceso (cambia a q_0) y se mueve a la derecha, por lo tanto:

$$\delta(q_2, y) = (q_2, y, L),$$
$$\delta(q_2, a) = (q_2, a, L),$$
$$\delta(q_2, x) = (q_0, x, R),$$

Ejemplo 2 MTAL V

4. Cuando todas las as se hayan procesado tendremos el siguiente movimiento:

$$\overbrace{xx\cdots x}^{n-1}q_2x\overbrace{yy\cdots y}^{n} \vdash \overbrace{xx\cdots x}^{n}q_0\overbrace{yy\cdots y}^{n},$$

es decir, la máquina tendrá que pasar a un nuevo estado q_3, respetar la y y moverse a la derecha. Por su parte, estando en q_3 y yendo hacia la derecha la máquina respeta todas las ys hasta encontrar el primer $\square$ en cuyo caso cambia a un nuevo estado q_4, respeta $\square$ y se mueve a la izquierda para llegar a la configuración final deseada $(\overbrace{xx\cdots x}^{n}\overbrace{yy\cdots y}^{n-1}q_f y)$, note que $q_f = q_4$.

$$\begin{aligned}\delta(q_0, y) &= (q_3, y, R),\\ \delta(q_3, y) &= (q_3, y, R),\\ \delta(q_3, \square) &= (q_4, \square, R).\end{aligned}$$

Ejemplo 2 MTAL VI

Juntando todo tenemos que $Q = \{q_0, q_1, q_2, q_3, q_4\}$, $F = \{q_4\}$ y

$$\begin{aligned}
\delta(q_0, a) &= (q_1, x, R), & \delta(q_2, y) &= (q_2, y, L),\\
\delta(q_0, y) &= (q_3, y, R), & \delta(q_2, a) &= (q_2, a, L),\\
\delta(q_1, a) &= (q_1, a, R), & \delta(q_2, x) &= (q_0, x, R),\\
\delta(q_1, y) &= (q_1, y, R), & \delta(q_3, y) &= (q_3, y, R),\\
\delta(q_1, b) &= (q_2, y, L), & \delta(q_3, \square) &= (q_4, \square, R).
\end{aligned}$$

Ejemplo 2 MTAL VI

Juntando todo tenemos que $Q = \{q_0, q_1, q_2, q_3, q_4\}$, $F = \{q_4\}$ y

$$\begin{aligned}
\delta(q_0, a) &= (q_1, x, R), & \delta(q_2, y) &= (q_2, y, L), \\
\delta(q_0, y) &= (q_3, y, R), & \delta(q_2, a) &= (q_2, a, L), \\
\delta(q_1, a) &= (q_1, a, R), & \delta(q_2, x) &= (q_0, x, R), \\
\delta(q_1, y) &= (q_1, y, R), & \delta(q_3, y) &= (q_3, y, R), \\
\delta(q_1, b) &= (q_2, y, L), & \delta(q_3, \square) &= (q_4, \square, R).
\end{aligned}$$

¿Qué pasa si tenemos una cadena de entrada de la forma $a^n b^m$ con $n < m$? Es decir, $m = n + k$, para $k > 1$.

Juntando todo tenemos que $Q = \{q_0, q_1, q_2, q_3, q_4\}$, $F = \{q_4\}$ y

$$\begin{aligned}
\delta(q_0, a) &= (q_1, x, R), & \delta(q_2, y) &= (q_2, y, L),\\
\delta(q_0, y) &= (q_3, y, R), & \delta(q_2, a) &= (q_2, a, L),\\
\delta(q_1, a) &= (q_1, a, R), & \delta(q_2, x) &= (q_0, x, R),\\
\delta(q_1, y) &= (q_1, y, R), & \delta(q_3, y) &= (q_3, y, R),\\
\delta(q_1, b) &= (q_2, y, L), & \delta(q_3, \Box) &= (q_4, \Box, R).
\end{aligned}$$

¿Qué pasa si tenemos una cadena de entrada de la forma $a^n b^m$ con $n < m$? Es decir, $m = n + k$, para $k > 1$.

$$\overbrace{xx\cdots x}^{n-1} q_2 x \overbrace{yy\cdots y}^{n} \overbrace{b\cdots b}^{k} \vdash \overbrace{xx\cdots x}^{n} q_0 \overbrace{yy\cdots y}^{n} \overbrace{b\cdots b}^{k} \vdash$$

$$\overbrace{xx\cdots x}^{n} y q_3 \overbrace{yy\cdots y}^{n-1} \overbrace{b\cdots b}^{k} \overset{n-1}{\vdash} \overbrace{xx\cdots x}^{n} \overbrace{yy\cdots y}^{n} q_3 \overbrace{b\cdots b}^{k}.$$

Ejemplo 2 MTAL VI

Juntando todo tenemos que $Q = \{q_0, q_1, q_2, q_3, q_4\}$, $F = \{q_4\}$ y

$$\begin{aligned}
\delta(q_0, a) &= (q_1, x, R), & \delta(q_2, y) &= (q_2, y, L),\\
\delta(q_0, y) &= (q_3, y, R), & \delta(q_2, a) &= (q_2, a, L),\\
\delta(q_1, a) &= (q_1, a, R), & \delta(q_2, x) &= (q_0, x, R),\\
\delta(q_1, y) &= (q_1, y, R), & \delta(q_3, y) &= (q_3, y, R),\\
\delta(q_1, b) &= (q_2, y, L), & \delta(q_3, \Box) &= (q_4, \Box, R).
\end{aligned}$$

¿Qué pasa si tenemos una cadena de entrada de la forma $a^n b^m$ con $n < m$? Es decir, $m = n + k$, para $k > 1$.

$$\overbrace{xx\cdots x}^{n-1} q_2 x \overbrace{yy\cdots y}^{n} \overbrace{b\cdots b}^{k} \vdash \overbrace{xx\cdots x}^{n} q_0 \overbrace{yy\cdots y}^{n} \overbrace{b\cdots b}^{k} \vdash$$

$$\overbrace{xx\cdots x}^{n} y q_3 \overbrace{yy\cdots y}^{n-1} \overbrace{b\cdots b}^{k} \overset{n-1}{\vdash} \overbrace{xx\cdots x}^{n} \overbrace{yy\cdots y}^{n} q_3 \overbrace{b\cdots b}^{k}.$$

Pero $\delta(q_3, b)$ no está definida así que la máquina para estando en el estado q_3 que no es estado final, por lo cual la palabra es rechazada.

Juntando todo tenemos que $Q = \{q_0, q_1, q_2, q_3, q_4\}$, $F = \{q_4\}$ y

$$
\begin{aligned}
\delta(q_0, a) &= (q_1, x, R), & \delta(q_2, y) &= (q_2, y, L),\\
\delta(q_0, y) &= (q_3, y, R), & \delta(q_2, a) &= (q_2, a, L),\\
\delta(q_1, a) &= (q_1, a, R), & \delta(q_2, x) &= (q_0, x, R),\\
\delta(q_1, y) &= (q_1, y, R), & \delta(q_3, y) &= (q_3, y, R),\\
\delta(q_1, b) &= (q_2, y, L), & \delta(q_3, \square) &= (q_4, \square, R).
\end{aligned}
$$

Ejemplo 2 MTAL VII

Juntando todo tenemos que $Q = \{q_0, q_1, q_2, q_3, q_4\}$, $F = \{q_4\}$ y

$$\begin{aligned}
\delta(q_0, a) &= (q_1, x, R), & \delta(q_2, y) &= (q_2, y, L),\\
\delta(q_0, y) &= (q_3, y, R), & \delta(q_2, a) &= (q_2, a, L),\\
\delta(q_1, a) &= (q_1, a, R), & \delta(q_2, x) &= (q_0, x, R),\\
\delta(q_1, y) &= (q_1, y, R), & \delta(q_3, y) &= (q_3, y, R),\\
\delta(q_1, b) &= (q_2, y, L), & \delta(q_3, \Box) &= (q_4, \Box, R).
\end{aligned}$$

¿Qué pasa si tenemos una cadena de entrada de la forma $a^n b^m$ con $n > m$? Es decir, $n = m + k$, para $k > 1$.

Ejemplo 2 MTAL VII

Juntando todo tenemos que $Q = \{q_0, q_1, q_2, q_3, q_4\}$, $F = \{q_4\}$ y

$$\begin{aligned}
\delta(q_0, a) &= (q_1, x, R), & \delta(q_2, y) &= (q_2, y, L),\\
\delta(q_0, y) &= (q_3, y, R), & \delta(q_2, a) &= (q_2, a, L),\\
\delta(q_1, a) &= (q_1, a, R), & \delta(q_2, x) &= (q_0, x, R),\\
\delta(q_1, y) &= (q_1, y, R), & \delta(q_3, y) &= (q_3, y, R),\\
\delta(q_1, b) &= (q_2, y, L), & \delta(q_3, \square) &= (q_4, \square, R).
\end{aligned}$$

¿Qué pasa si tenemos una cadena de entrada de la forma $a^n b^m$ con $n > m$? Es decir, $n = m + k$, para $k > 1$.

$$\overbrace{x\cdots x}^{m-1} q_2 x \overbrace{a\cdots a}^{k}\overbrace{y\cdots y}^{m} \vdash \overbrace{x\cdots x}^{m} q_0 \overbrace{a\cdots a}^{k}\overbrace{y\cdots y}^{m} \vdash$$

$$\overbrace{x\cdots x}^{m+1} q_1 \overbrace{a\cdots a}^{k-1}\overbrace{y\cdots y}^{m} \overset{k-1+m}{\vdash} \overbrace{x\cdots x}^{m+1}\overbrace{a\cdots a}^{k-1}\overbrace{y\cdots y}^{m} q_1 \square.$$

Ejemplo 2 MTAL VII

Juntando todo tenemos que $Q = \{q_0, q_1, q_2, q_3, q_4\}$, $F = \{q_4\}$ y

$$
\begin{aligned}
\delta(q_0, a) &= (q_1, x, R), & \delta(q_2, y) &= (q_2, y, L),\\
\delta(q_0, y) &= (q_3, y, R), & \delta(q_2, a) &= (q_2, a, L),\\
\delta(q_1, a) &= (q_1, a, R), & \delta(q_2, x) &= (q_0, x, R),\\
\delta(q_1, y) &= (q_1, y, R), & \delta(q_3, y) &= (q_3, y, R),\\
\delta(q_1, b) &= (q_2, y, L), & \delta(q_3, \square) &= (q_4, \square, R).
\end{aligned}
$$

¿Qué pasa si tenemos una cadena de entrada de la forma $a^n b^m$ con $n > m$? Es decir, $n = m + k$, para $k > 1$.

$$
\overbrace{x \cdots x}^{m-1} q_2 x \overbrace{a \cdots a}^{k} \overbrace{y \cdots y}^{m} \vdash \overbrace{x \cdots x}^{m} q_0 \overbrace{a \cdots a}^{k} \overbrace{y \cdots y}^{m} \vdash
$$

$$
\overbrace{x \cdots x}^{m+1} q_1 \overbrace{a \cdots a}^{k-1} \overbrace{y \cdots y}^{m} \overset{k-1+m}{\vdash} \overbrace{x \cdots x}^{m+1} \overbrace{a \cdots a}^{k-1} \overbrace{y \cdots y}^{m} q_1 \square.
$$

Pero $\delta(q_1, \square)$ no está definida, así que la máquina para estando en el estado q_1 que no es estado final, por lo cual la palabra es rechazada.

El cálculo que hace la máquina con la palabra de entrada $aabb$ es el siguiente:

$$\begin{aligned} q_0aabb &\vdash xq_1abb \vdash xaq_1bb \vdash xq_2ayb \\ &\vdash q_2xayb \vdash xq_0ayb \vdash xxq_1yb \\ &\vdash xxyq_1b \vdash xxq_2yy \vdash xq_2xyy \\ &\vdash xxq_0yy \vdash xxyq_3y \vdash xxyyq_3\Box \\ &\vdash xxyy\ \Box q_4\Box. \end{aligned}$$

Ejemplo 2 MTAL VIII

El cálculo que hace la máquina con la palabra de entrada $aabb$ es el siguiente:

$$
\begin{aligned}
q_0aabb &\vdash xq_1abb \vdash xaq_1bb \vdash xq_2ayb \\
&\vdash q_2xayb \vdash xq_0ayb \vdash xxq_1yb \\
&\vdash xxyq_1b \vdash xxq_2yy \vdash xq_2xyy \\
&\vdash xxq_0yy \vdash xxyq_3y \vdash xxyyq_3\square \\
&\vdash xxyy\ \square q_4\square .
\end{aligned}
$$

Es decir, la máquina para estando en q_4 que es el estado final, por lo tanto la palabra es aceptada.

Máquinas de Turing como transductores

- Además de que las máquinas de Turing pueden aceptar lenguajes, también pueden actuar como transductores, es decir, dispositivos que transforman su entrada en salida.

Máquinas de Turing como transductores

- Además de que las máquinas de Turing pueden aceptar lenguajes, también pueden actuar como transductores, es decir, dispositivos que transforman su entrada en salida.
- La entrada serán todos los símbolos consecutivos distintos de $\square$ que estén en la cinta al inicio del cálculo.

Máquinas de Turing como transductores

- Además de que las máquinas de Turing pueden aceptar lenguajes, también pueden actuar como transductores, es decir, dispositivos que transforman su entrada en salida.
- La entrada serán todos los símbolos consecutivos distintos de $\square$ que estén en la cinta al inicio del cálculo.
- La salida serán los símbolos consecutivos distintos de $\square$ que se encuentren en la cinta cuando la máquina pare.

Máquinas de Turing como transductores

- Además de que las máquinas de Turing pueden aceptar lenguajes, también pueden actuar como transductores, es decir, dispositivos que transforman su entrada en salida.
- La entrada serán todos los símbolos consecutivos distintos de $\square$ que estén en la cinta al inicio del cálculo.
- La salida serán los símbolos consecutivos distintos de $\square$ que se encuentren en la cinta cuando la máquina pare.

Así podemos ver a una máquina de Turing transductora M como la implementación de una función f definida mediante

$$\widehat{w} = f(w)$$

siempre que

$$q_0 w \overset{*}{\vdash}_M q_f \widehat{w},$$

para algún estado final q_f.

Definición 11

Una función f con dominio D se dice que es **Turing computable** o sólo **computable** si existe alguna máquina de Turing $M = (Q, \Sigma, \Gamma, \delta, q_0, \Box, F)$ tal que

$$q_0 w \overset{*}{\vdash}_M q_f f(w), q_f \in F,$$

para todo $w \in D$.

Ejemplo la suma es Turing computable I

Ejemplo 12

Dados dos enteros positivos x y y, diseñe una máquina de Turing que calcule $x + y$.

Solución:

1. Por simplicidad, usaremos notación unaria para representar a los enteros positivos. Concretamente un entero positivo x se representa mediante $w(x) \in \{1\}^+$, tal que $|w(x)| = x$.

Ejemplo la suma es Turing computable I

Ejemplo 12

Dados dos enteros positivos x y y, diseñe una máquina de Turing que calcule $x + y$.

Solución:

2. Asumiremos que $w(x)$ y $w(y)$ están en la cinta separados por un 0 y que la cabeza de lectura-escritura está sobre el símbolo más a la izquierda de $w(x)$.

Ejemplo la suma es Turing computable I

Ejemplo 12

Dados dos enteros positivos x y y, diseñe una máquina de Turing que calcule $x + y$.

Solución:

3. Cuando el cálculo termine, $w(x + y)$ estará en la cinta seguida seguida por un 0 y la cabeza de lectura-escritura estará sobre el símbolo más a la izquierda de $w(x + y)$.

Ejemplo la suma es Turing computable II

Por lo tanto, queremos diseñar una máquina de Turing que realice el siguiente cálculo:

$$q_0 w(x) 0 w(y) \overset{*}{\vdash} q_f w(x+y) 0, q_f \in F.$$

Como estamos usando notacion unaria, lo único que hay que hacer es "mover" el 0 que separa a $w(x)$ y $w(y)$ al final de $w(x+y)$, regresar la cabeza al inicio de $w(x+y)$ y movernos a un estado final.

Ejemplo la suma es Turing computable III

1. Al inicio estando en q_0 la cabeza de lectura-escritura leerá un 1, entonces la máquina permanecerá en q_0, escribirá un 1 y se moverá a la derecha hasta encontrar el único 0 de la cadena, en cuyo caso cambiara a q_1 cambiará el 0 por 1 y se moverá a la derecha.

$$\delta(q_0, 1) = (q_0, 1, R),$$
$$\delta(q_0, 0) = (q_1, 1, R),$$

Ejemplo la suma es Turing computable III

2. Estando en q_1 tendrá que moverse a la derecha "saltando" los 1s de $w(y)$, hasta que encuentre un $\square$, en cuyo caso lo respeta, cambia a un nuevo estado q_2 y se mueve a la izquierda.

$$\delta(q_1, 1) = (q_1, 1, R),$$
$$\delta(q_1, \square) = (q_2, \square, L),$$

Ejemplo la suma es Turing computable IV

3. Estando en q_2 va a leer el último 1 de $w(y)$ lo cambia a 0, con lo cual habremos "movido" al final de $w(x+y)$ el cero que separaba a $w(x)$ y $w(y)$, pasa a un nuevo estado q_3 y se mueve a la izquierda.

$$\delta(q_2, 1) = (q_3, 0, L).$$

Ejemplo la suma es Turing computable IV

4. Estando en q_3 y yendo hacia la izquierda la máquina de Turing "salta" todos los 1s de $w(x+y)$ hasta que encuentra el $\square$ que está a la izquierda de $w(x+y)$, en cuyo caso lo respeta pasa al estado final q_4 y se mueve a la derecha para quedar en la configuración final deseada.

$$\delta(q_3, 1) = (q_3, 1, L),$$
$$\delta(q_3, \square) = (q_4, \square, R).$$

Ejemplo la suma es Turing computable V

Juntando todo tenemos que $Q = \{q_0, q_1, q_2, q_3, q_4\}$, $F = \{q_4\}$ y

$$\begin{aligned} \delta(q_0, 1) &= (q_0, 1, R), & \delta(q_2, 1) &= (q_3, 0, L), \\ \delta(q_0, 0) &= (q_1, 1, R), & \delta(q_3, 1) &= (q_3, 1, L), \\ \delta(q_1, 1) &= (q_1, 1, R), & \delta(q_3, \square) &= (q_4, \square, R). \\ \delta(q_1, \square) &= (q_2, \square, L), \end{aligned}$$

El cálculo que hace la máquina con $w(x) = 111$ y $w(y) = 11$ es el siguiente:

$$\begin{aligned} q_0 111011 &\vdash 1q_0 11011 \vdash 11q_0 1011 \vdash 111q_0 011 \\ &\vdash 1111q_1 11 \vdash 11111q_1 1 \vdash 111111q_1 \square \\ &\vdash 11111q_2 1 \vdash 1111q_3 10 \vdash 111q_3 110 \\ &\vdash 11q_3 1110 \vdash 1q_3 11110 \vdash q_3 111110 \\ &\vdash q_3 \square 111110 \vdash q_4 111110 = q_4 w(x+y)0. \end{aligned}$$

Ejemplo 13

Diseñe una máquina de Turing que copie cadenas de 1s. Más precisamente, construya una máquina que realice el siguiente cálculo:

$$q_0 w \overset{*}{\vdash} q_f w w$$

Solución:

1. Por cada 1 de w debemos de poner otro 1.
2. Pero debemos saber cuántos 1s hemos procesado, así que se deben cambiar primero todos los 1s por algún otro marcador, digamos x.
3. Luego cambiamos la x más derecha por 1 y vamos a la parte más derecha de la zona de celdas que no son blancas y ahí sustituimos el primer $\square$ por 1.
4. Así procedemos hasta que no queden más xs.

Ejemplo copiar cadenas de 1s es computable II

1. Al inicio estando en q_0 lee un 1, lo cambia por x y continúa así hasta que todos los 1s han sido cambiados por x, en ese momento pasa al nuevo estado q_1.

$$\delta(q_0, 1) = (q_0, x, R),$$
$$\delta(q_0, \square) = (q_1, \square, L).$$

2. Estando en q_1 debe moverse a la izquierda saltando los posibles 1s que encuentre hasta llegar a la x más derecha, en cuyo caso la cambia por un 1 y pasa a un nuevo estado q_2. Note que si ya se procesaron todas las xs, estando en q_1 se encontrará el $\square$ que está a la izquierda de la porción de celda no blanca, en cuyo caso pasa al estado final q_3 y se mueve a la derecha.

$$\delta(q_1, 1) = (q_1, 1, L),$$
$$\delta(q_1, x) = (q_2, 1, R),$$
$$\delta(q_1, \square) = (q_3, \square, R).$$

Ejemplo copiar cadenas de 1s es computable III

3. Estando en q_2 salta los 1s que encuentre yendo hacia la derecha, hasta que encuentra el primer $\Box$, lo cambia por 1, regresa el control a q_1 (para que busque la siguiente x más derecha o termine).

$$\delta(q_2, 1) = (q_2, 1, R),$$
$$\delta(q_2, \Box) = (q_1, 1, L).$$

Veamos un cálculo con $w = 11$.

$$\begin{aligned} q_0 11 &\vdash x q_0 1 \vdash x x q_0 \Box \vdash x q_1 x \\ &\vdash x 1 q_2 \Box \vdash x q_1 11 \vdash q_1 x 11 \\ &\vdash 1 q_2 11 \vdash 11 q_2 1 \vdash 111 q_2 \Box \\ &\vdash 11 q_1 11 \vdash 1 q_1 111 \vdash q_1 1111 \\ &\vdash q_1 \Box 1111 \vdash q_3 1111. \end{aligned}$$

Ejemplo comparar enteros positivos es computable

Ejemplo 14

Sean x y y dos enteros positivos representados en notación unaria. Construya una máquina de Turing que pare en un estado final q_y si $x \geq y$ y que pare en un estado no final q_n si $x < y$. Específicamente, una máquina que realice el siguiente cálculo:

$$q_0 w(x) 0 w(y) \overset{*}{\vdash} q_y w(x) 0 w(y) \text{ si } x \geq y,$$

$$q_0 w(x) 0 w(y) \overset{*}{\vdash} q_n w(x) 0 w(y) \text{ si } x < y.$$

Solución: Adapte la solución del Ejemplo 10. En lugar de empatar as con bs, empate los 1s a la izquierda del 0 con los 1s a la derecha. Al final del empate tenemos tres casos.

$$x \cdots x 0 x \cdots x, \text{ si } x = y,$$

$$x \cdots x 1 \cdots 1 0 x \cdots x, \text{ si } x > y,$$

$$x \cdots x 0 x \cdots x 1 \cdots 1, \text{ si } x < y.$$

Tesis de Turing

Tesis 15

Cualquier cálculo que se pueda realizar por medios mecánicos puede calcularse mediante alguna máquina de Turing.

Definición de algoritmo

Definición 16

Un **algoritmo** para una función $f : D \to R$ es una máquina de Turing M, a la que se le da como entrada en su cinta cualquier $d \in D$ y eventualmente para con la respuesta correcta $f(d) \in R$ sobre su cinta. Específicamente, se requiere que:

$$q_0 d \overset{*}{\vdash}_M q_f f(d), q_f \in F, \text{ para todo } d \in D.$$

Ejercicios

1. Construya máquinas de Turing que acepten los siguientes lenguajes sobre $\{a, b\}$, verifique sus máquinas con dos ejemplos.
   1. $L = \{w : |w| \text{ es impar}\}$.
   2. $L = \{w : |w| \text{ es múltiplo de } 4\}$.
   3. $L = \{w : n_a(w) \neq n_b(w)\}$.
   4. $L = \{a^n b^n a^n b^n : n \neq 0\}$.
   5. $L = \{a^n b^{2n} : n \geq 1\}$.
2. Construya la máquina de Turing especificada en el Ejemplo 14, verifique su máquina con dos ejemplos.
3. Construya máquinas de Turing que calculen las siguientes funciones para x y y enteros positivos representados en unario, verifique sus máquinas con dos ejemplos.
   1. $f(w) = w^R$, donde $w \in \{0, 1\}^+$.
   2. $f(x) = 2x + 1$.
   3. $f(x, y) = 2x + 3y$.
   4. $f(x) = x \mod 5$.

Fin del curso
Muchas gracias