Autómatas y lenguajes formales

Víctor Mijangos de la Cruz

vmijangosc @ ciencias.unam.com

III. Máquinas de Turing

Máquinas de Turing

Máquina de Turing

Máquina de Turing

Una máquina de Turing es una 7-tupla $M = (Q, \Sigma, \Gamma, \delta, q_0, B, F)$ tal que:

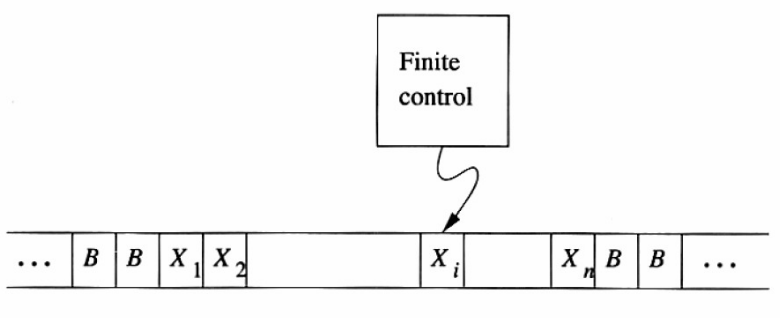
1. Q es un conjunto finito de estados.
2. Σ es un conjunto finito de símbolos de entrada.
3. Γ es un conjunto de símbolos de la cinta, con $\Sigma \subseteq \Gamma$.
4. $\delta : Q \times \Gamma \to Q \times \Gamma^2$ es la función de transición, que toma $\delta(q, X) = (p, Y, D)$, donde $q, p \in Q$, $X, Y \in \Gamma$ y D es una dirección L (izquierda) o R (derecha).
5. $q_0 \in Q$ es el estado inicial, y $F \subseteq Q$ es conjunto de estados finales.
6. $B \in \Gamma$ es el símbolo espacial 'blanco', tal que $B \notin \Sigma$.

Elementos de Máquina de Turing

Una máquina de Turing contará con los siguientes elementos:

1. **Control finito:** Corresponde a los elementos $Q, \Sigma, q_0, \delta, F$; es decir, es el elemento que controla en qué estado se encuentra actualmente la máquina.
2. **Cinta:** Corresponde a los símbolos Γ, incluyendo B; se divide en celdas o cuadros.
3. **Celda:** Es un elemento de la cinta que puede contener cualquier símbolo del conjunto finito Γ.
4. **Entrada:** Es una cadena de símbolos en Σ^*.
5. **Símbolos de la cinta:** Son los símbolos que se encuentran en las celdas de la cinta.
6. **Cabezal de cinta:** Es un elemento que se posiciona en una de las celdas de la cinta. Decimos que la máquina **escanea** esa celda. El cabezal inicia a la izquierda de la cinta.
7. **Movimiento:** Los movimientos de la máquina de Turing son: a) cambio de estado; b) escribir un símbolo en la celda escaneada; y c) mover el cabezal a la izquierda o derecha de la cinta.

Visualización de máquina de Turing



El control finito puede verse como un autómata finito, pero supeditado a la lectura de una cinta.

La cinta tiene infinitas celdas: sólo una sección de la cinta tiene símbolos de $\Gamma \setminus \{B\}$.

Hay un número infinito de símbolos B a la izquierda y la derecha de la cinta.

Descripción instantánea

Una descripción instantánea (DI) determina el estado del control finito, así como los símbolos que el cabezal escanea en la cinta. Esta descripción está dada como:

1. Cada DI cuenta con un número infinito de prefijos y sufijos que son las celdas que no han sido visitadas.
2. Los símbolos B (blancos) no se colocan, de tal forma que la descripción es una cadena finita.
3. Se posiciona el estado a la izquierda del símbolo que el cabezal está escaneando.

La estructura de una descripción instantánea es:

$$\cdots BX_1X_2\cdots X_{i-1}qX_iX_{i+1}\cdots X_nB\cdots$$

Donde cada $X_j \in \Gamma$ son los símbolos en las celdas de la cinta; q es el estado del control finito actual, y X_i es el símbolo que escanea el cabezal.

Notación $\vdash$

$\vdash$

Dada una máquina de Turing M, representaremos sus transiciones como: dado $\delta(q, X_i) = (p, Y, L)$, tendremos que:

$$X_1 X_2 \cdots X_{i-1} q X_i \cdots X_n \vdash_M X_1 X_2 \cdots p X_{i-1} Y \cdots X_n$$

Esto es:

1. Se cambia el estado q por el estado p (control finito).
2. El símbolo escaneado X_i se reemplaza por Y en la cinta.
3. El cabezal se mueve a la izquierda L para escanear el símbolo anterior. Si es R (derecha) se mueve hacia el símbolo siguiente.

Excepciones a la notación

Si $\delta(q, X_i) = (p, Y, L)$, movimiento a la izquierda, se dan 2 excepciones:

1. Si $i = 1$, entonces entonces $qX_1X_2 \cdots X_n \vdash_M pBX_1X_2 \cdots X_n$
2. Si $i = n$ y $Y = B$, entonces $X_1X_2 \cdots X_{n-1}qX_n \vdash_M X_1X_2 \cdots pX_{n-1}$

Si $\delta(q, X_i) = (p, Y, R)$, movimiento a la izquierda, se dan 2 excepciones:

1. Si $i = n$, entonces $X_1X_2 \cdots X_{n-1}qX_n \vdash_M X_1X_2 \cdots X_{n-1}X_npB$
2. Si $i = 1$ y $Y = B$, entonces entonces $qX_1X_2 \cdots X_n \vdash_M pX_2 \cdots X_n$

Tabla de transiciones

Tabla de transiciones

Dada una máquina de Turing $M = (Q, \Sigma, \Gamma, \delta, q_0, B, F)$, podemos representar las transiciones a partir de una tabla, tal que:

1. Los renglones son los estados posibles del control finito.
2. Las columnas son los símbolos de Γ, comenzando por los símbolos en Σ y terminando con el símbolo B.

La tabla será de la forma:

	a_1	$\cdots$	a_k	X_1	$\cdots$	X_m	B
q_0	$\delta(q_0, a_1)$	$\cdots$	$\delta(q_0, a_n)$	$\delta(q_0, X_1)$	$\cdots$	$\delta(q_0, X_m)$	$\delta(q_0, B)$
$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
q_n	$\delta(q_n, a_1)$	$\cdots$	$\delta(q_n, a_n)$	$\delta(q_n, X_1)$	$\cdots$	$\delta(q_n, X_m)$	$\delta(q_n, B)$

Por tanto, cada entrada tendrá (p, Y, D); si no existe transición, no se colocará nada o se colocará $-$.

Ejemplo

Defínase una máquina de Turing $M = (Q, \Sigma, \Gamma, \delta, q_0, B, F)$ con $Q = \{q_i : i = 0, 1, 2, 3, 4\}$, $\Sigma = \{0, 1\}$, $\Gamma = \{0, 1, X, Y, B\}$, $F = \{q_4\}$ y δ dada por la tabla:

	0	1	X	Y	B
q_0	(q_1, X, R)	-	-	(q_3, Y, R)	-
q_1	$(q_1, 0, R)$	(q_2, Y, L)	-	(q_1, Y, R)	-
q_2	$(q_2, 0, L)$	-	(q_0, X, R)	(q_2, Y, L)	-
q_3	-	-	-	(q_3, Y, R)	(q_4, B, R)
q_4	-	-	-	-	-

Esta máquina de Turing realiza su cómputo con la cinta conteniendo cadenas de la forma $X^*0^*Y^*1^*$, por lo que lee cadenas de la forma 0^*1^*

Ejemplo

De la máquina de Turing M anterior, tenemos la siguiente descricpión para la cadena de entrada

$$0011$$

La descripción es:

$$\begin{aligned} q_0 0011 &\vdash Xq_1 011 \vdash X0q_1 11 \vdash Xq_2 0Y1 \\ &\vdash q_2 X0Y1 \vdash Xq_0 0Y1 \vdash XXq_1 Y1 \\ &\vdash XXYq_1 1 \vdash XXq_2 YY \vdash Xq_2 XYY \\ &\vdash XXq_0 YY \vdash XXYq_3 Y \vdash XXYYq_3 B \\ &\vdash XXYYBq_4 \end{aligned}$$

M termina en el estado q_4 que es terminal, por lo que ha consumido la cadena de entrada llegando a un estado terminal.

Ejemplo

La máquina de Turing puede no consumir toda la cadena de entrada y/o quedarse en un estado en el que no pueda avanzar. Considérese la cadena de entrada

$$010$$

Tenemos la descripción:

$$\begin{aligned} q_0 010 &\vdash Xq_1 10 \vdash q_2 XY0 \vdash Xq_0 Y0 \\ &\vdash XYq_3 0 \end{aligned}$$

Como no hay una regla definida para $\delta(q_3, 0)$, M no puede seguir y se detiene en un estado no final, y sin consumir la cadena de entrada.

Diagramas de transición

Al igual que los AFD y los PDA, las máquinas de Turing pueden ilustrarse en Diagramas:

Diagrama de transición

Un diagrama de transición para una máquina de Turing $M = (Q, \Sigma, \Gamma, \delta, q_0, B, F)$, su diagrama de transición consiste en:

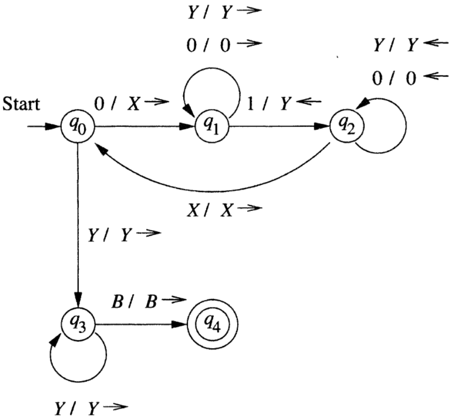
1. Un conjunto de nodos asociados a los estados Q de M.
2. Arcos que van de $q \in Q$ hacia $p \in Q$ etiquetados de la forma, dado $\delta(q, X) = (p, Y, D)$:

$$X/YD$$

3. El estado inicial se indica con una flecha y el final con doble círculo.

Ejemplo de diagrama

De la máquina de Turing anterior para el lenguaje 0^*1^*, tenemos el diagrama:



Máquinas de Turing para computar funciones

Una máquina de Turing también puede utilizarse para **computar funciones**, esto se puede hacer de la siguiente forma:

- El argumento de la función es la cadena de entrada, por ejemplo, una función en enteros se puede representar por un dígito binario.
- El resultado de la función es lo que la máquina escribe en la cinta al terminar de procesar la cadena de entrada.

Función Monus

La función monus, o substracción propia, se puede definir como

$$m \dot{-} n = \max\{m - n, 0\}$$

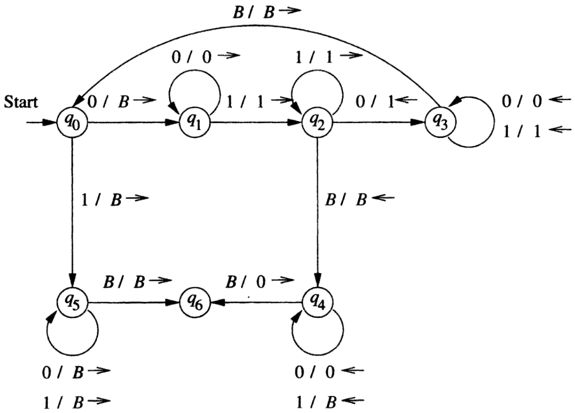
Construimos una máquina de Turing $M = (Q, \Sigma, \Gamma, \delta, q_0, B, F)$ que computa esta función de la siguiente forma:

1. Los estados son $Q = \{q_0, q_1, q_2, q_3, q_4, q_5, q_6\}$, con inicial q_0, y el alfabeto de entrada $\Sigma = \{0, 1\}$.
2. Los símbolos de la cinta son $\Gamma = \{0, 1, B\}$ con B el símbolo blanco.
3. Se considerará un estado final $F = \{q_6\}$.
4. La **entrada** serán cadenas de la forma $0^m 1 0^n$ y la **salida**, lo escrito en la cinta al final, $0^{m \dot{-} n}$.

Función Monus

Para computar la función definimos la siguiente tabla de transición:

	0	1	B
q_0	(q_1, B, R)	(q_5, B, R)	-
q_1	$(q_1, 0, R)$	$(q_2, 1, R)$	-
q_2	$(q_3, 1, L)$	$(q_2, 1, R)$	(q_4, B, L)
q_3	$(q_3, 0, L)$	$(q_3, 1, L)$	(q_0, B, R)
q_4	$(q_4, 0, L)$	(q_4, B, L)	$(q_6, 0, R)$
q_5	(q_5, B, R)	(q_5, B, R)	(q_6, B, R)
q_6	-	-	-



Función Monus (Ejemplo)

Para $3 \dot{-} 2$, la entrada es 000100, por lo que se tiene:

$$
\begin{aligned}
q_0000100 &\vdash q_100100 \vdash 0q_10100 \vdash 00q_1100 \vdash 001q_200 \\
&\vdash 00q_3110 \vdash 0q_30110 \vdash q_300110 \vdash q_3B00110 \\
&\vdash q_000110 \vdash q_10110 \vdash 0q_1110 \vdash 01q_210 \\
&\vdash 011q_20 \vdash 01q_311 \vdash 0q_3111 \vdash q_30111 \\
&\vdash q_3B0111 \vdash q_00111 \vdash q_1111 \vdash 1q_211 \\
&\vdash^* 111q_2B \vdash 11q_41 \vdash 1q_41 \vdash q_41 \vdash q_4B \\
&\vdash 0q_6
\end{aligned}
$$

Intuición sobre aceptación

De forma intuitiva, una máquina de Turing acepta una cadena cuando se tienen los siguientes pasos:

1. La cadena de entrada se coloca en la cinta.
2. El cabezal comienza desde el símbolo que se encuentra hasta la izquierda.
3. Si la máquina de Turing entra eventualmente en el estado de aceptación, la cadena es aceptada.

Aceptación de máquina de Turing

Lenguaje de una máquina de Turing

Sea $M = (Q, \Sigma, \Gamma, \delta, q_0, B, F)$ una máquina de Turing, el lenguaje de M se define como:

$$L(M) = \{w \in \Sigma^* : \exists q_f \in F, \exists \alpha, \beta \in \Gamma^*, q_0 w \vdash_M^* \alpha q_f \beta\}$$

Lenguaje recursivamente enumerable

Un lenguaje L es recursivamente enumerable (RE) si existe una máquina de Turing M tal que $L = L(M)$.

Detención

Detención

Decimos que una máquina de Turing se detiene si entra en un estado $q \in Q$ tal que $\delta(q, X)$ no esta definido para algún $X \in \Gamma$.
Es decir, la máquina de Turng ya no puede realizar ningún movimiento.

Ejemplo: En el caso de la máquina de Turing para la función monus, vemos que el estado de detención es q_6. También en q_0 y q_1 se detendrá con el símbolo B.
Por ejemplo, con una cadena como 100, tenemos que:

$$q_0 100 \vdash^* Bq_6B$$

Suposición: Se asume que una máquina de Turing siempre se detiene cuando está en un estado de aceptación.

Tipos de Máuinas de Turing

Almacenamiento en los estados

Control finito con almacenamiento

El control finito con almacenamiento constará de los siguientes elementos:

1. El estado control $q \in Q$.
2. El almacenamiento, que corresponde a uno o más símbolos de cinta.

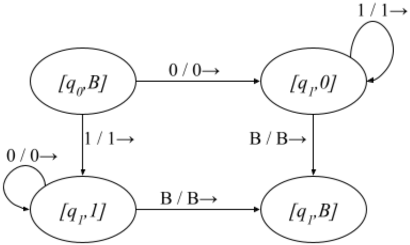
El control finito será una tupla de la forma:

$$[q, X_1, X_2, ..., X_n]$$

Donde q es el estado actual y X_i son símbolos almacenados.

Ejemplo

Piénsese en la máquina de Turing $M = (Q, \{0, 1\}, \{0, 1, B\}, [q_0, B], B, \{[q_1, B]\})$, donde $Q \subseteq \Sigma \times \Gamma$.



	0	1	B
$[q_0, B]$	$([q_1, 0], 0, R)$	$([q_1, 1], 1, R)$	-
$[q_1, 0]$	-	$([q_1, 0], 1, R)$	$([q_1, B], B, R)$
$[q_1, 1]$	$([q_1, 1], 0, R)$	-	$([q_1, B], B, R)$
$[q_1, B]$	-	-	-

Ejemplo

Lo que hace esta máquina de Turing es alamcenar en cada estado el símbolo primero. Por ejemplo, para la cadena: 011

$$[q_0, B]011 \vdash 0[q_1, 0]11 \vdash 01[q_1, 0]1 \vdash 011[q_1, 0]B$$
$$\vdash 011B[q_1, B] \in F$$

El lenguaje que acepta esta MT es $L(M) = [[01^* + 10^*]]$, además observamos que se detiene, por ejemplo con una cadena no aceptada como 010:

$$[q_0, B] \vdash 0[q_1, 0]10 \vdash 01[q_1, 0]0!$$

Multidimensionalidad

La cinta de una máquina de Turing puede pensarse como conteniendo diferentes pistas, de tal forma que se pueda escribir en diferentes pistas en una cinta.

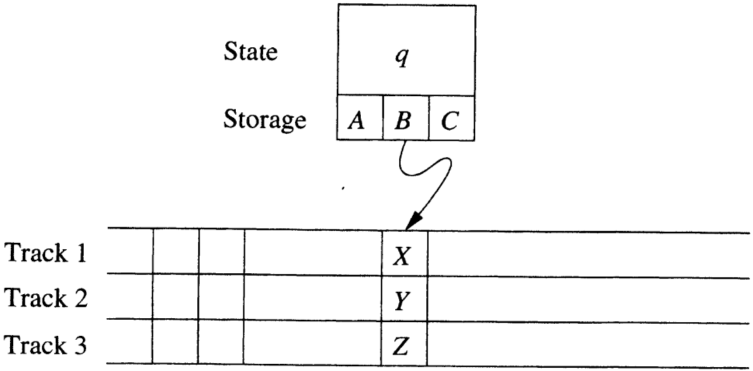
Máquina de Turing multidimensiona

Una máquina de Turing multidimensional $M = (Q, \Sigma, \Gamma, \delta, q_0, B, F)$ cuenta con múltiples pistas en su cinta, si su cinta toma símbolos de $\Gamma \times \cdots \Gamma$; esto es, la escritura en la cinta es de la forma:

$$[X_1, X_2, \cdots X_n]$$

Máquina de Turing multidimencional y con almacenamiento

Una máquina de Turing puede contener almacenamiento en su control finito y multiples pistas, para conformar una máquina más compleja.



Subrutinas

Subrutina

Una subrutina de una máquina de Turing es un conjunto de estados $Q' \subseteq Q$ que ejecutan algunos (sub-)procesos de utilidad. Este conjunto incluye:

1. Un estado inicial.
2. Un estado de `return`, el cual pasa el control de la subtutina a la máquina de Turing.

Dada una máquina de Turing, una subrutina será **llamada** cuando haya una transición a su estado inicial.

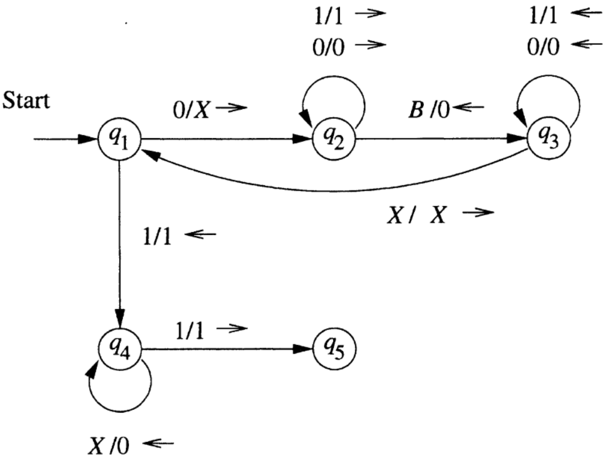
Ejemplo

Diseñamos una máquina de Turing que compute la función de **multiplicación**, con las siguientes especificaciones:

- La entrada de la cinta es de la forma $0^m 1 0^n 1$ para realizar el producto $m \cdot n$.
- La salida será de la forma $0^{m \cdot n}$.
- El **paso básico** consiste en cambiar un 0 del primer grupo a B y añadir n 0's al último grupo.
- Esto implica copiar el grupo de n 0's en el final m veces; $0^{i-1} 1 0^n 1 0^{(k+1)n}$.
- Finalmente, los elementos que no son del bloque final se pasan a blancos.

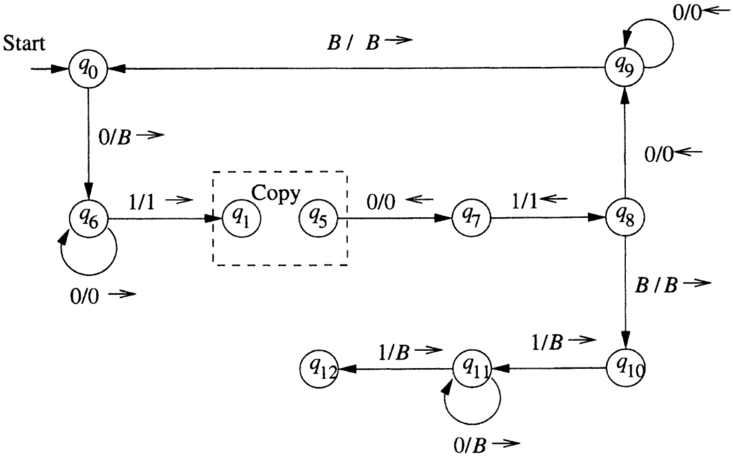
Ejemplo (subrutina copiar)

Definimos una subrutina para la acción de `copiar` que esta dada por el siguiente diagrama:



Ejemplo (Incorporación de subrutina)

La máquina de Turing para la multiplicación está dada como:



Ejemplo (Aplicación)

Podemos usar la máquina de Turing para multiplicar 2 por 3. La entrada será 0010001:

$$
\begin{aligned}
q_0 0010001 &\vdash Bq_6 010001 \vdash 0q_6 10001 \vdash 01q_1 0001 \\
\texttt{Subrutina} &\vdash 01Xq_2 001 \vdash^* 01X001q_2 \vdash 01X00q_3 10 \vdash^* 01q_3 X0010 \\
&\vdash 01XXq_2 0100 \vdash^* 01XX010q_2 \vdash 01XX010q_3 0 \vdash^* 01XXq_1 0100 \\
&\vdash 01XXXq_2 100 \vdash^* 01XXX100q_2 \vdash 01XXX100q_3 0 \vdash^* 01XXXq_1 1000 \\
&\vdash 01XXq_4 X1000 \vdash 01Xq_4 X01000 \vdash^* 0q_4 10001000 \\
&\vdash 01q_5 0001000 \qquad \texttt{Return}
\end{aligned}
$$

La subrutina se ejecuta cuando se entra a q_1 y retorna el valor en q_5, como se ve, lo que hace es copiar n número de 0's al final de la cadena.

Ejemplo (Multiplicación)

Una vez que vemos cómo funciona la subrutina, podemos obviarla en el proceso de la multiplicación:

$$
\begin{aligned}
q_0 0010001 &\vdash Bq_6 010001 \vdash 0q_6 10001 \vdash 01q_1 0001 \\
&\vdash \texttt{Subrutina} \vdash 01q_5 0001000 \vdash^* q_8 010001000 \vdash q_9 B010001000 \\
&\vdash q_0 010001000 \vdash Bq_6 10001000 \vdash 1q_1 0001000 \vdash \texttt{Subrutina} \\
&\vdash 1q_5 0001000000 \vdash q_7 10001000000 \vdash q_8 B10001000000 \vdash q_{10} 10001000000 \\
&\vdash q_{11} 0001000000 \vdash^* q_{11} 1000000 \vdash q_{12} 000000
\end{aligned}
$$

Por tanto la salida es $000000 = 0^6$ pues $6 = 2 \cdot 3$.

Máquina de Turing con múltiples cintas

Máquina de Turing con múltiples cintas

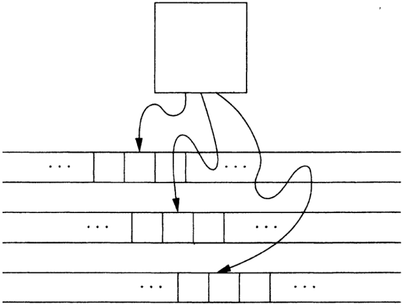
En una máquina de Turing con múltiples cintas está determinada por los siguientes elementos:

1. Entrada: La secuencia finita de símbolos con las que comienza la cinta.
2. En las otras celdas sólo se cuenta con símbolos blancos.
3. El control finito se encuentra en principio en el estado inicial.
4. El cabezal de la primera cinta está a la izquierda de la entrada.
5. Todos los otros cabezales se ubican en celdas arbitrarias de sus respectivas cintas.

Movimientos en las múltiples cintas

Los movimientos de la máquina de Turing con múltiples cintas dependen del símbolo escaneado por los cabezales:

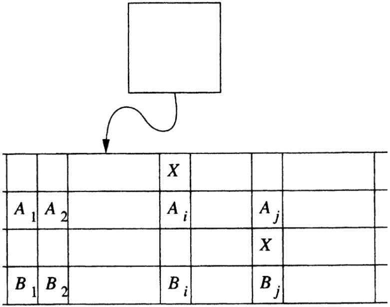
1. El control finito entra en un nuevo estado.
2. En cada cinta se escribe un nuevo símbolo en la celda escaneada.
3. Cada cabezal hace un movimientos, L o R o estacionario. Los cabezales se mueven de manera independiente.



Equivalencia con máquina de Turing

Teorema

Una máquina de Turing con múltiples cintas puede ser emulada por una máquina de Turing (multidimensional) en tiempo $O(n^2)$.



Supóngase que $L = L(M)$ donde M es una máquina de Turing con k cintas. Considérese a N, una máquina de Turing con $2k$ pistas.

De estas pistas, k tienen las cintas de M, y las k restantes indican con una marca la posición de los cabezales.

De tal forma, que N puede simular a M observando la posición de cada cabezal y los símbolos de estas en sus pistas. Por tanto $L = L(N)$.

Máquinas de Turing no deterministas

Máquinas de Turing no-determinista

Una máquina de Turing no-determinista (MTN) es una 7-tupla $M = (Q, \Sigma, \Gamma, \delta, q_0, B, F)$ en donde la función de transición está definida como:

$$\delta : Q \times \Gamma \to \mathcal{P}(Q \times \Gamma^2)$$

Es decir, las funciones de transición son de la forma:

$$\delta(q, X) = \{(q_i, Y_i, D_i) : i \in \{1, 2, ..., k\}\}$$

Lenguaje de una MTN

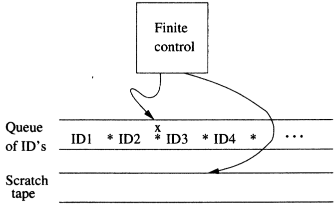
Una máquina de Turing no-determinista acepta una cadena $w \in \Sigma^*$ si hay una secuencia de movimientos desde el estado inicial hasta el final consumiendo w.

Teorema

Teorema

Si M_N es una máquina de Turing no-determinista, entonces existe una máquina de Turing determinista M_D tal que $L(M_N) = L(M_D)$.

Definamos una máquina de Turing determinista M_D con múltiples cintas. La primera cinta guardará de descripciones inmediatas de M_N. Una descripción de M_N se separan $*$.



Demostrración (continuación)

Para procesar el ID actual, M_D hará los siguiente:

1. M_D examina el estado y símbolo de ID. Dentro del control finito de M_D están las elecciones de M_N para cada símbolo. Si el estado del ID es de aceptación, entonces M_D acepta.
2. Si el estado no acepta, y la combinación de símbolos tiene k movimientos, entonces M_D copia el ID en la segunda cinta y hace k copias de ese ID al final de la secuencia de ID's en cinta 1.
3. M_D modifica los k ID's de acuerdo a una diferencia de las k elecciones de movimientos que M_N tiene desde el ID actual.
4. M_d regresa a la marca del ID actual, la borra y se mueve a la marca del siguiente ID a la derecha. Entonces repite el ciclo.

Al final, se comprueba que $L(M_N) = L(M_D)$.

Cinta semi-infinita

Máquina de Turing con cinta semi-infinita

Una máquina de Turing se dice que tiene cinta semi-infinita si su cinta sólo contiene celdas a la derecha de la posición inicial y no hay celdas en la izquierda.

Restricción de blanco

La restricción del blanco restringe a una máquina de Turing para que nunca escriba un blanco. Esto es $\nexists q, p, X, M$ tal que $\delta(q, X) = (p, B, M)$

Teorema

Todo lenguaje aceptado por una máquina de Turing es también aceptado por una máquina de Turing con las restriciones: 1) cuenta con cinta semi-infinita; y 2) nunca escribe un blanco.

Máquina de Turing Universal

Modelo de una computadora

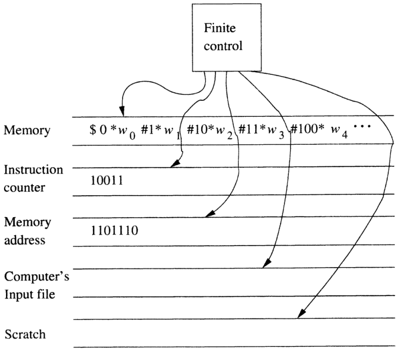
El modelo de una computadora cuenta con los siguientes elementos:

1. El almacenamiento corresponde a una secuencia indefinidamente larga de **palabras**, cada una con una **dirección**:
    1. Las palabras cuentan con 32 o 64 bits.
    2. Las direcciones son enteros positivos. Las palabras tienen direcciones múltiplos de 4 o 8.
2. El **programa** se almacena en una palabra de memoria.
3. Una **instrucción** implica un número finito de palabras. Cada instrucción cambia el contenido de una palabra.
4. Los **Registros** son palabras de memoria con acceso rápido.

Simulación de una computadora

La simulación de una computadora se hará por medio de una máquina de Turing con los siguientes elementos:

1. **Cinta de memoria**: Representa la memoria completa de la computadora tal que:
   1. Las direcciones de las palabras de memoria alternan con los contenidos de palabras de memoria.
   2. Las direcciones usan los símbolos $*$ y $\#$ para encontrarse. EL símbolo \$ indica inicio de secuencia.
2. **Conteo de instrucciones**: Contiene un entero en binario que representa locación de memoria de la cinta 1. Es la siguiente instrucción a ejecutarse.
3. **Memoria de dirección**: Es el contenido de la dirección después que se ha localizado en la cinta 1.



Ciclos de instrucciones

Las instrucciones del programa de computadora se ejecutarán de la siguiente forma:

1. Se busca en la cinta 1 (memoria) la dirección que se encuentra en la cinta 2 (conteo).
2. Se examina el valor de la dirección: los primeros bits de una instrucción representan la acción, y los restantes la dirección envuelta en la acción.
3. Si la instrucción requiere un valor de dirección, se copia la dirección en la cinta 3 (memoria de dirección). Se marca la posición de instrucción en cinta 4 (scratch).
4. Se ejecutan las instrucciones, algunas de estas pueden ser:
    1. Copiar alguna dirección.
    2. Sumar valores de direcciones.
    3. Saltar, i.e., ir a otra instrucción.
5. Después de la acción, sumar 1 al contador de instrucciones (cinta 2) y comenzar el ciclo de nuevo.

Equivalencia de MT y computadoras

Lema

Si una computadora cuenta sólo con: 1) instrucciones que aumentan la longitud máxima de la palabra a lo mucho en 1; y 2) instrucciones que una máquina de Turing de múltiples cintas puede ejecutar en palabras de longitud k en $O(k^2)$ pasos, entonces, la máquina de Turing simula n pasos de la computadora en $O(n^3)$ pasos.

Teorema

Una computadora del tipo que hemos descrito anteriormente puede simularse, con n pasos, por una máquina de Turing de una cinta en $O(n^6)$ pasos.

Indexación de cadenas binarias

Isomorfismo para indexación

Definimos el isomorfismo $f\colon \{0,1\}^* \to \mathbb{N}$ como:

- $f(\epsilon) = 0$, $f(0) = 1$ y $f(1) = 2$
- Si $w \in \{0,1\}^*$ y $x \in \{0,1\}$, entonces $f(wx) = 2f(w) + f(x)$.

Nuestros objetivos son:

1. Asignar un código a cada máquina de Turing posible.
2. Mostrar que existe un número infinito numerable de máquinas de Turing.
3. Definir una máquina de Turing universal, que simule cada una de estas máquinas de Turing particulares.

Códigos para máquinas de Turing

Dada una máquina de Turing $M = (Q, \Sigma, \Gamma, \delta, q_1, B, F)$ con alfabeto binario $\Sigma = \{0, 1\}$, que siempre se detenga, considérese:

1. Dado $|Q| = r$, sea q_1 estado inicial y $q_2 \in F$ el único final.
2. Dado $\Gamma = \{X_1, X_2, ..., X_s\}$, sean $X_1 = 0, X_2 = 1$ y $X_3 = B$.
3. Considérense las direcciones $L = D_1$ y $R = D_2$.

Codificación de transición: Dada $\delta(q_i, X_j) = (q_k, X_l, D_m)$ con $i, j, k, l, , m \in \mathbb{N}$, su codificación será:

$$C = 0^i 1 0^j 1 0^k 1 0^l 1 0^m$$

Codificación de máquina de Turing: Dada la máquina de Turing M con transiciones codificadas como $C_1, ..., C_n$, la codificación de M será:

$$M = C_1 11 C_2 11 \cdots 11 C_{n-1} 11 C_n$$

Ejemplo

Considérese la máquina de Turing M con los siguientes elementos:

1. $Q = \{q_1, q_2, q_3\}$, $\Sigma = \{0, 1\}$, $\Gamma = \{0, 1, B\}$
2. Las transiciones y sus códigos son: $\delta(q_1, 1) = (q_3, 0, R)$,

Transición	**Código**	**Transición**	**Código**
$\delta(q_1, 1) = (q_3, 0, R)$	0100100010100	$\delta(q_3, 0) = (q_1, 1, R)$	0001010100100
$\delta(q_3, 1) = (q_2, 0, R)$	00010010010100	$\delta(q_3, B) = (q_3, 1, L)$	0001000100010010

Por lo que el código de esta máquina de Turing es el siguiente:

*0100100010100**11**0001010100100**11**00010010010100**11**0001000100010010*

Lenguaje universal

Lenguaje universal

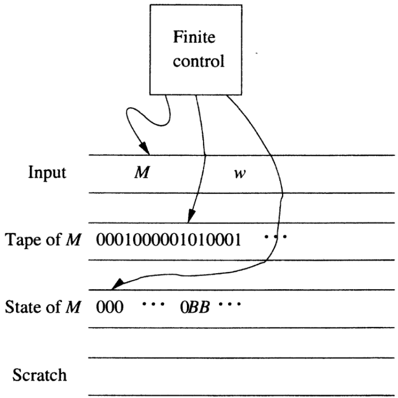
Definimos al lenguaje universal, L_U, como el conjunto de cadenas binarias que codifican los pares (M, w) tal que M es el código de una máquina de Turing y $w \in \{0, 1\}^*$ es una cadena binaria.

- El lenguaje universal codifica una máquina de Turing y sus posibles entradas.
- Si bien las entradas de la máquina de Turing pueden estar en un alfabeto Σ arbitrario, siempre se puede codificar en alfabeto binario.
- Nuestro propósito es definir una máquina de Turing que acepte este lenguaje universal L_U; es decir, que emule a cualquier otra máquina de Turing.

Máquina de Turing Universal

Máquina de Turing Universal

Una máquina de Turing universal es una máquina de Turing U tal que $L(U) = L_U$.



Para construir la máquina de Turing Universal, requerimos de las siguientes cintas:

1. **Cinta 1**: Almacena las transiciones de otra máquina de Turing M, junto con la cadena w.
2. **Cinta 2**: Simula la cinta de M por medio del formato para codificar M. El símbolo X_i se representa como 0.
3. **Cinta 3**: Guarda los estados de M en el formato de la codificación; el estado q_j se codifica como 0^j.

Operación de máquina de Turing Universal

U opera de la siguiente forma:

1. Se examina la entrada en la cinta 1. Si no se reconoce como el código de una máquina de Turing, U se detiene sin aceptar.
2. La cinta 2 se inicializa con w en su forma codificada ($0 \mapsto 10$ y $1 \mapsto 100$).
3. En la cinta 3, se pone 0 (inicial de M) y se mueve el cabezal de U en la cinta 2 a la primera celda de la simulación.
4. En la cinta 1, se busca la transición de M ($0^i10^j10^k10^l10^m$), tal que 0^i es estado en cinta 3 y 0^j símbolo en la cinta 2. Los movimientos se simulan como:
    1. Transición $p_i \to p_k$: se cambia el contenido de la cinta 3 de 0^i a 0^k.
    2. Cambio de símbolo X_j por X_l: en la cinta 2 se cambia 0^j por 0^l.
    3. Mover el cabezal: si $m = 1$ el cabezal de la cinta 2 se mueve a la izquierda; si $m = 2$ se mueve a la derecha.

Operación de máquina de Turing Universal

U aceptará o rechazará una entrada según los siguientes casos:

1. Si M no tiene transiciones que coincidan con el estado y símbolo en U, U se detiene, pues M lo hace.
2. Si M está en estado de aceptación, U estará en estado de aceptación también.

Teorema

Si (M, w) es un par que codifica a una máquina de Turing y su entrada, y U una máquina de Turing Universal, entonces U acepta (M, w) si y sólo si M acepta w.

Corolario

Una máquina de Turing Universal puede simular cualquier otra máquina de Turing.

Tesis Church-Turing

Tesis de Turing

Para cualquier sistema formal determinista, existe una máquina de Turing formalmente equivalente.

Efectivamente computable

Decimos que un problema es efectivamente calculable si podemos producir para éste un resultado deseado.

Tesis de Church-Turing

Un problema sobre conjuntos discretos (números naturales) es efectivamente computable si y sólo si es computable por una máquina de Turing.

La tesis de Church-Turing nos dice que los problemas efectivamente calculables son los mismos que los problemas computables por una máquina de Turing.