Máquinas de Turing

Autómatas y lenguajes formales

6 de noviembre de 2022

Máquina de Turing

La máquina de Turing es uno de los modelos de computación más poderosos, es lo más cercano a una computadora (como la que usaron para bajar este pdf, y quizá hasta para leerlo), pero no es el único modelo. Tomaremos la máquina de Turing como el formalismo con el que trabajaremos ahora, para entender las cosas, pero más adelante veremos otros modelos de máquinas. No sé si será el método más pedagógico, o el más sencillo, pero es el estándar en la mayoría de libros del tema (además de que Turing es el más famoso de los matemáticos que trabajaron inicialmente en el área). Los modelos son equivalentes, así que no perderemos generalidad.

A su vez hay variaciones en los modelos de máquinas de Turing cuyas características extras ayudan a hacer más entendible el modelo, reducir la notación, ahorrar un poco de trabajo, pero de nueva cuenta, son equivalentes entre sí.

Una máquina de Turing se compone de un conjunto finito de estados Q, una cinta semi-infinita limitada por la izquierda con el símbolo $\vdash$ e ilimitada por la derecha (este límite izquierdo es para saber donde empieza la cinta) y una cabeza que puede moverse a izquierda y derecha, capaz de leer y escribir caracteres en la cinta[1]. Las palabras de entrada, de longitud finita, se escriben sobre la cinta de izquierda a derecha (como escribimos nosotros). Al terminar la palabra de entrada en el resto de casillas de la cinta para distinguir que no contienen caracter alguno está pre-escrito el símbolo ␣. Un esquema ejemplificando esta disposición se muestra en la figura **??**.

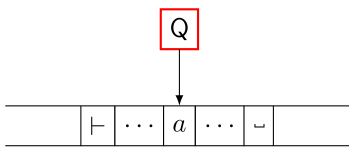

Figura 1: Esquema mecánico de una máquina de Turing.

[1]Son muy jóvenes para recordar los casetes quizá, aunque por ahí hay unos intentos retros de revivirlos, pero si alguna vez han visto uno con su respectivo reproductor podrán notar que cuenta con una cinta magnética de color café o negro, que es leída por una cabeza también magnética. Era un formato de no muy buena calidad, de riesgo pues un imán de potencia suficiente podía dañar la cinta y los carretes podían provocar accidentes como enrollarse o atorarse. Hubo computadoras que usaban casetes para leer programas, para una crónica al respecto echen un ojo a: `https://www.xataka.com/historia-tecnologica/cuando-los-videojuegos-venian-en-cassette-y-habia-que-rebobinarlos-para-poder-jugar`

Esto puede verse un poco más complicado que nuestro clásico diagrama de estados, aquí los estados no son tan obvios ya que además van acompañados de acciones sobre la cabeza lectora y sobre la cinta. La máquina se encontrará en su estado inicial, leerá el carácter en la cinta y dependiendo de lo que lea y el estado escribe un nuevo símbolo (puede escribir el mismo símbolo y se entiende que no alteró la cinta) y la cabeza lectora se mueve a la izquierda o derecha en una sola posición, una vez completado esto entra en un nuevo estado.

Las reglas que nos dan las transiciones entre estados son las ya conocidas transiciones δ. Para detenerse la máquina entra en su estado final de *aceptación* o *rechazo*, aunque existe la posibilidad de que no entre a ninguno de estos estados y se quede trabajando por siempre, en un *loop* o ciclo infinito.

Definición formal

Definición 1 *Una máquina de Turing determinista, de cinta única es una 9-tupla (yo lo traduciría como un noneto o enéada, pero quizá no es la terminología) descrita como:*

$$M = (Q, \Sigma, \Gamma, \vdash, \llcorner\!\lrcorner, \delta, s, t, r)$$

con:

- Q *es el conjunto finito de estados*
- Σ *es el alfabeto de entrada (finito)*
- Γ *es el alfabeto de cinta (finito), con* $\Sigma \subseteq \Gamma$
- $\vdash \in \Gamma - \Sigma$ *el símbolo de inicio de la cinta*
- $\llcorner\!\lrcorner \in \Gamma - \Sigma$ *el símbolo de espacio en blanco*
- $\delta : Q \times \Gamma \to Q \times \Gamma \times \{\to, \leftarrow\}$ *la función de transición*
- $s \in Q$ *el estado inicial*
- $t \in Q$ *el estado de aceptación*
- $r \in Q$ *el estado de rechazo.*

La manera de describir las transiciones es: $\delta(p, a) = (q, b, \to)$, que se entiende como "si la maquina en el estado p lee en la posición de la cinta en que se encuentra la cabeza lectora una a entonces pasa a un estado q, escribe en la misma posición de la cinta una b y mueve la cabeza lectora a la derecha ($\to$ o R)". Un caso especial es cuando se lee el símbolo $\vdash$, la cinta no puede ir más a la izquierda, entonces si se tiene: $\delta(q, \vdash) = (p, \vdash, \to)$, nunca con $\leftarrow$.

Si se entra a los estados de aceptación o rechazo, ahí se queda la máquina: $\delta(t, a) = (t, b, \to / \leftarrow)$, puede moverse izquierda o derecha de la cinta, incluso escribir en la cinta pero ya no sale del estado, y $\delta(r, d) = (r, f, \to / \leftarrow)$, de manera similar, puede moverse en cualquier dirección de la cinta la cabeza, puede escribir en ella, pero no sale del estado de rechazo[2].

[2]Incluso puede ponerse $\delta(t, a) = (t, -, -)$ o $\delta(r, a) = (t, -, -)$ para indicar que ya no importa que se haga, al entrar al estado de aceptación o rechazo ya no importa lo que pase después

Un ejemplo

Como hemos hecho regularmente vamos a ver un ejemplo que vaya más allá de los modelos que hemos visto antes, en este caso trabajaremos el lenguaje dependiente de contexto $\{a^n b^n c^n | n \geq 0\}$. Lo que hará la máquina es lo siguiente:

0 Empieza en su estado inicial, supongamos que la cabeza lectora está al inicio de la cinta (si no lo está solo haría falta poner una función de *reset* o algo parecido, no es relevante ahora, imaginemos todo ideal), al leer que la cinta está en su orilla izquierda la cabeza lectora empieza a moverse a la derecha.

1 Seguimos en el estado inicial y lo que esperamos al movernos al primer símbolo no vacío de la cinta es hallar una a, mientras se lean a's la cabeza sigue moviéndose a la derecha, se sobrescriben las a's y se queda en el estado inicial.

2 Al leer la primera b se pasa al estado q_1 y la cabeza lectora sigue andando a la derecha, mientras lea b's se quedará en el estado q_1 y la cabeza seguirá corriendo a la derecha.

3 La máquina sigue en el estado q_1 y lee una c, entonces se pasa al estado q_2 y la cabeza lectora sigue su camino a la derecha, mientras lea c's se queda en el estado q_2.

3b Si está en el estado q_1 leyendo b's y de repente se encuentra una a en lugar de una c, se manda al estado de rechazo r y la cinta podría seguir corriendo, pero no tiene sentido, se puede parar ahí, fin, la cadena no es parte del lenguaje.

4 La máquina se encuentra en el estado q_2 leyendo c's y se encuentra con un símbolo vacío, es decir $\llcorner\!\lrcorner$, entonces pasa al estado q_3 y escribe un símbolo especial, en este caso escribimos una $\dashv$ pero podría ser una x, o un símbolo que no vayamos a usar, este no es crucial, sólo es para que en los próximos recorridos sepa donde pararse. Hasta aquí sólo se hizo el primer recorrido, se comprobó que la cadena tiene el orden correcto en los caracteres, ahora hay que contar. Una vez llegando al final, marcando la orilla derecha, la cabeza empieza a viajar a la izquierda.

5 Fase de conteo: La cabeza ahora corre hacia la izquierda (de regreso), si encuentra una c la borra (escribiendo $\llcorner\!\lrcorner$) y pasa al estado q_4, mientras la cabeza sigue corriendo a la izquierda. Las siguientes c's que encuentre las sobre escribirá, se quedará en el mismo estado y seguirá la cabeza corriendo a la izquierda.

5b Si está en el estado q_3, corriendo a la izquierda la cabeza lectora, y lee una b o a en lugar de una c, inmediatamente se va al estado de rechazo (en principio no debería pasar, pues ya checamos el orden en la primera fase, pero nunca está de más reiterar las órdenes, además de que en muchos casos no se guarda memoria de instrucciones anteriores).

6 La máquina está en el estado q_4 y corre a la izquierda, si de repente se encuentra una b, se borra (se sustituye por $\llcorner\!\lrcorner$), se pasa al estado q_5 y la cabeza

	$\vdash$	a	b	c	$\sqcup$	$\dashv$
s	$(s, \vdash, \rightarrow)$	$(s, a, \rightarrow)$	$(q_1, b, \rightarrow)$	$(q_2, c, \rightarrow)$	$(q_3, \dashv, \leftarrow)$	
q_1		$(r, -, -)$	$(q_1, b, \rightarrow)$	$(q_2, c, \rightarrow)$	$(q_3, \dashv, \leftarrow)$	
q_2		$(r, -, -)$	$(r, -, -)$	$(q_2, c, \rightarrow)$	$(q_3, \dashv, \leftarrow)$	
q_3	$(t, -, -)$	$(r, -, -)$	$(r, -, -)$	$(q_4, \sqcup, \leftarrow)$	$(q_3, \sqcup, \leftarrow)$	
q_4	$(r, -, -)$	$(r, -, -)$	$(q_5, \sqcup, \leftarrow)$	$(q_4, c, \leftarrow)$	$(q_4, \sqcup, \leftarrow)$	
q_5	$(r, -, -)$	$(q_6, \sqcup, \leftarrow)$	$(q_5, b, \leftarrow)$		$(q_5, \sqcup, \leftarrow)$	
q_6	$(q_7, \vdash, \rightarrow)$	$(q_6, a, \leftarrow)$			$(q_6, \sqcup, \leftarrow)$	
q_7		$(q_8, \sqcup, \rightarrow)$	$(r, -, -)$	$(r, -, -)$	$(q_7, \sqcup, \rightarrow)$	$(t, -, -)$
q_8		$(q_8, a, \rightarrow)$	$(q_9, \sqcup, \rightarrow)$	$(r, -, -)$	$(q_8, \sqcup, \rightarrow)$	$(r, -, -)$
q_9			$(q_9, b, \rightarrow)$	$(q_{10}, \sqcup, \rightarrow)$	$(q_9, \sqcup, \rightarrow)$	$(r, -, -)$
q_{10}				$(q_{10}, c, \rightarrow)$	$(q_{10}, \sqcup, \rightarrow)$	$(q_3, \dashv, \leftarrow)$

Cuadro 1: Tabla de transiciones de la máquina de Turing.

continúa su camino a la izquierda. Mientras encuentre b subsecuentes se quedará en el mismo estado y no les hará nada a esos símbolos (reescribirá b), la cabeza seguirá su camino a la izquierda.

6b Si se encuentra en el estado q_4 y en lugar de b la cabeza lectora lee una a o el símbolo del extremo izquierdo de la cinta, $\vdash$, se va a un estado de rechazo y no hace nada más.

7 La máquina se encuentra en el estado q_5, la cabeza corre a la izquierda, si en su camino encuentra una a la sustituye por $\sqcup$, pasa al estado q_6 y continúa corriendo a la izquierda. Las siguientes a's que encuentre las reescribirá, se quedará en el mismo estado y seguirá la cabeza su camino a la izquierda.

7b Si se encuentra en el estado q_5 y en lugar de a la cabeza lectora lee la orilla izquierda de la cinta, $\vdash$, se va a un estado de rechazo y no hace nada más.

8 En el estado q_6, la cabeza sigue corriendo a la izquierda, en este punto ya ha marcado (borrando) una c, una b y una a, si la cabeza lee el símbolo $\vdash$ lo reescribe, pasa al estado q_7 y la cabeza se da vuelta, empieza a correr a la derecha y de nueva cuenta borra conforme se vaya encontrando en orden una a, una b y una c.

La idea es que la cinta vaya recorriendo, borrando una y otra vez, si se encuentra símbolos sin borrar cuando los correspondientes en orden ya hayan sido borrados, rechaza la cadena, si no halla ya más que vacío, acepta la cadena. Hay algunas características que no definí pero pueden verse en la tabla 1 de la página 4.

Hay que ver más ejemplos, por el momento les dejo definiciones importantes:

Definición 2 *Un lenguaje es* Turing-reconocible *si alguna máquina de Turing lo reconoce. También se dice que es recursivamente enumerable.*

La colección de cadenas aceptadas por una máquina de Turing T, es el lenguaje reconocido por la máquina, $L(T)$.

Definición 3 *Un lenguaje es* Turing-decidible *o simplemente* decidible *si alguna máquina de Turing lo decide. También se dice que es recursivo.*

¿Cuál es la diferencia entre decidible o reconocible? Una cadena reconocida es aceptada por la máquina de Turing, está en su alfabeto de entrada, pero puede ser que la cadena provoque que la máquina entre a un estado de aceptación, a uno de rechazo o se quede atrapada en un ciclo infinito.

Quedar atrapado en un ciclo infinito es un problema, a veces no se puede saber si está en un ciclo infinito o sólo está tardando demasiado (pasa en la vida), entonces nos gustaría tener máquinas de Turing que sólo puedan entrar a estados de aceptación o rechazo, nada de ciclos infinitos (esto es muy idóneo), en ese caso, las cadenas que al entrar en una máquina de Turing sólo pueden producir estados de aceptación o rechazo, nunca ciclos infinitos, forman un lenguaje Turing-decidible, y la máquina asociada se llama total. Determinar estos lenguajes es un gran problema teórico en la mayoría de máquinas.

Una forma matemática de decir si una cadena es decidible:

$$(s, \vdash \alpha \llcorner\!\lrcorner^{\infty}, 0) \implies {}^{*}(q, \vdash \beta \llcorner\!\lrcorner^{\infty}, n)$$

q debe ser el estado t (aceptación) o r (rechazo), n son los pasos que toma para llegar al estado de aceptación o rechazo. Lo que verá en la cinta antes de empezar será $\vdash \alpha \llcorner\!\lrcorner^{\infty}$, con $\alpha \in \Sigma^*$ y los símbolos que ya conocemos, es una forma reducida de decir qué está escrito en la cinta, como es infinita y la cadena aceptada es finita, los espacios en blanco al final de la cinta los escribimos como $\llcorner\!\lrcorner^{\infty}$. Al terminar la lectura y escritura y llegar a un estado de aceptación o rechazo, seguramente habrá alterado la cinta (como hicimos en el ejemplo anterior) y ahora hay una nueva cadena $\beta \in \Gamma^*$, en caso de que hallamos agregado símbolos nuevos como indicadores.

Otro ejemplo

Veamos un ejemplo más, sea la máquina de Turing que decida el lenguaje $A = \{0^{2^n} | n \geq 0\}$, es decir, el lenguaje con puras cadenas de 0's que su longitud sea una potencia de 2. Sea nuestra máquina M y la haremos algo parecida a la del ejemplo anterior, la idea es llevar la cuenta de la cantidad de 0's que hay.

1. Estamos en el estado inicial s, si hay un sólo 0 la idea es aceptarlo (es parte del lenguaje), pero aún hay que comprobar que es el único. Para los siguientes ceros se llevará la cuenta, por el momento, para evitar un movimiento extra hasta el borde izquierdo de la cinta el primer 0 lo podemos sustituir por un espacio en blanco $\llcorner\!\lrcorner$ y movemos la cabeza lectora a la derecha buscando más 0's, pasamos del estado inicial al estado q_1. Si la cadena está vacía o tiene cualquier otro símbolo, vamos al estado de rechazo r.

2. Estamos en q_1 y se lee un espacio en blanco, es decir, se acaba la cadena, entonces hay un sólo 0 y se va a el estado de aceptación t. El estado q_1 será crucial para todas las cadenas aceptadas, hay que tenerlo en cuenta. Si en cambio se lee otro 0, la cabeza escribe una x sobre él, empezamos a llevar la cuenta. Cambiamos al estado q_2 y se mueve la cabeza lectora a la derecha. Si se lee una x quiere decir que ya estamos en una de las

subsecuentes vueltas de la cabeza, ya llevamos ese 0 en la cuenta, nos quedamos en el mismo estado y la cabeza lectora se mueve a la derecha (a la izquierda ya no hay a qué ir, hay un espacio en blanco).

3. Vamos en el estado q_2, si estamos aquí es que llevamos al menos dos ceros detrás de nosotros, el siguiente 0 que se reciba no se marca (se reescribe el 0), pasamos al estado q_3 y la cabeza lectora se mueve a la derecha. Si en lugar de un 0 se lee un espacio en blanco ␣ quiere decir que llegamos a la orilla derecha de la cinta, entonces se pasa al estado q_4, no se altera el ␣ y la cabeza empieza a moverse a la izquierda (vamos de regreso). Si se lee una x es que ya pasamos por aquí al menos una vez, entonces ya se agregó ese cero a la cuenta, no se cambia lo escrito en la cinta, se queda en el mismo estado y la cabeza lectora sigue su camino a la derecha.

4. Estamos en el estado q_3, si la cabeza lectora encuentra un 0 lo marca, es decir, lo sustituye por una x, regresa al estado q_2 y la cabeza continúa su recorrido a la derecha. Entre estos dos estados entra en un ciclo, dejando un 0 sin marcar y marcando el siguiente, así hasta que llega al final de la cinta o sucede lo siguiente: estamos en el estado q_3 y la cabeza lectora encuentra el final de la cadena, eso quiere decir que llevamos un número impar de 0's, en definitiva no puede ser de longitud potencia de dos si es impar, se va al estado r, si encuentra una casilla marcada x quiere decir que ya pasamos por aquí en un recorrido anterior, no se hace nada (reescribe x), se queda en el mismo estado y continúa su camino a la derecha.

5. Estamos en el estado q_4, recordemos que llegamos aquí porque en el estado q_3 dimos con el final derecho de la cadena, la cabeza ya corre hacia la izquierda y empieza a recorrer lo que ya recorrimos al menos una vez, los 0's y x's que encuentre los deja sin alterar (los reescribe) y continúa su camino a la izquierda, se queda en el mismo estado q_4, pero en el momento que encuentre la orilla izquierda de la cinta (que recordarán es un 0 que marcamos como ␣) entonces regresa al estado q_1 sin cambiar nada de ese limite de la cinta y la cabeza vuelve a viajar hacia la derecha, vamos darle un repasada a la cadena. Del estado q_2 decide si debe seguir contando (marcando con x) o si ya está todo marcado y se llega al final derecho de la cinta, la cadena pasa al estado t.

Escribimos la tabla con estas transiciones:

	$\vdash$	0	x	␣
s	$(s, \vdash, \to)$	$(q_1, ␣, \to)$	$(r, -, -)$	$(r, -, -)$
q_1		$(q_2, x, \to)$	$(q_1, x, \to)$	$(t, -, -)$
q_2		$(q_3, 0, \to)$	$(q_2, x, \to)$	$(q_4, ␣, \leftarrow)$
q_3		$(q_2, x, \to)$	$(q_3, x, \to)$	$(r, -, -)$
q_4		$(q_4, 0, \leftarrow)$	$(q_4, x, \leftarrow)$	$(q_1, ␣, \to)$

Para t (aceptación) y r (rechazo) no hay transiciones, no las ponemos, los guiones dentro de los estados es porque ya no importa que pase con la cinta y la cabeza lectora, y los espacios en blanco son estados inaccesibles. Como no son tantos estados, podemos hacer un diagrama de estados, como se ve en la figura 2, noten la diferencia entre $\Rightarrow$ que hace referencia a la transición, y $\to$ que es

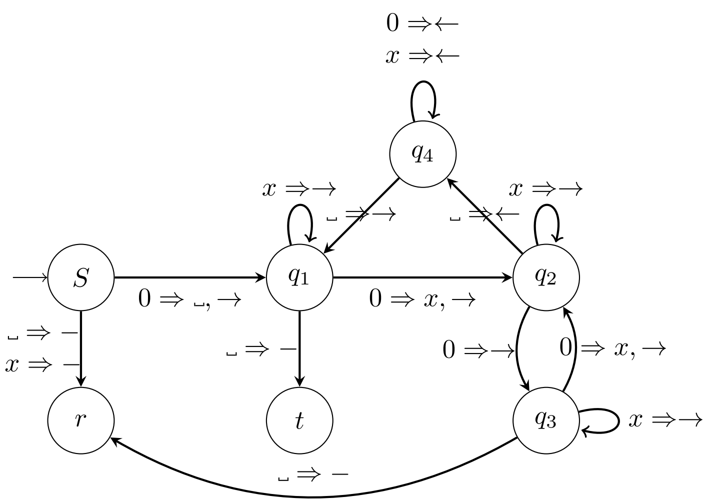

Figura 2: Diagrama de estados de la máquina de Turing descrita.

el movimiento hacia la derecha de la cabeza lectora, como ya se mencionó $-$ denota que ya no importa a dónde se mueva la cabeza lectora.

Esta construcción puede parecer un tanto sacada de la manga ¿cómo nos consta que funciona para el lenguaje mencionado? Pues debemos checar, por ejemplo la cadena 0000, de acuerdo a la descripción está en el lenguaje, vamos a meterla en nuestra máquina para checar. Lo vamos viendo por pasos:

$\overrightarrow{s}0000$ Empezamos la maquina en el estado s y en la orilla izquierda de la cinta. La cabeza lectora empezará a correr hacia la derecha, donde se encuentra el primer cero, el cual cambiará por un ␣ y pasará al estado q_1.

$␣\overrightarrow{q_1}000$ Noten que los estados van recorriendo la cadena, pero no ocupan espacios en la cinta, la ponemos ahí para facilidad de entendimiento. Estamos en el estado q_1, la cabeza lectora corre hacia la derecha, donde halla un 0, lo cambia por un x (lo marca), pasa al estado q_2.

$␣x\overrightarrow{q_2}00$ Ya en el estado q_2 la cabeza sigue corriendo a la derecha donde encuentra otro cero, como se puede ver en el diagrama y la tabla de transiciones, en este estado de hallarse un 0 no lo marca, se va al estado q_3 y la cabeza sigue corriendo a la derecha.

$␣x0\overrightarrow{q_3}0$ Ahora en el estado q_3 la cabeza lectora corriendo hacia la derecha se encuentra un cero, por la tabla de transiciones cambia el 0 por una x, regresa al estado q_2 y la cinta sigue corriendo hacia la derecha.

$␣x0x\overrightarrow{q_2}␣$ Llegamos al final de la palabra (no habíamos puesto antes el símbolo de espacio vacío pero realmente ahí estaba al final de la cadena), estamos en el estado q_2, cuando esto pasa se va al estado q_4, deja igual escrito el espacio vacío y la cabeza lectora empieza a correr ahora hacia la izquierda (eso lo tendremos en cuenta en nuestras próximas cadenas).

$\textvisiblespace x0x\overleftarrow{q_4}\textvisiblespace$ Ahora estamos en el estado q_4, la cabeza lectora está corriendo a la izquierda, así que se encuentra una x, no la altera, se queda en el mismo estado y la cabeza sigue corriendo a la izquierda.

$\textvisiblespace x0\overleftarrow{q_4}x\textvisiblespace$ Seguimos en el estado q_4, la cabeza lectora corre hacia la izquierda y se encuentra un 0, lo deja igual, se queda en el mismo estado y la cabeza lectora sigue corriendo a la izquierda.

$\textvisiblespace x\overleftarrow{q_4}0x\textvisiblespace$ Seguimos en el estado q_4, la cabeza lectora corre hacia la izquierda y se encuentra una x, la deja igual, se queda en el mismo estado y la cabeza lectora sigue corriendo a la izquierda.

$\textvisiblespace\overleftarrow{q_4}x0x\textvisiblespace$ En el estado q_4, la cabeza corre hacia la izquierda y se encuentra el espacio vacío del inicio de la cadena (el primer cero que convertimos en un $\textvisiblespace$), entonces pasa al estado q_1, la cabeza empieza a correr a la derecha y deja el espacio como estaba.

$\textvisiblespace\overrightarrow{q_1}x0x\textvisiblespace$ Ahora en el estado q_1 la cabeza corriendo a la derecha se halla una x (un cero que marcamos en un paso anterior), se queda en el mismo estado, la cabeza corre a la derecha y no se cambia nada en la cinta.

$\textvisiblespace x\overrightarrow{q_1}0x\textvisiblespace$ Estamos en el estado q_1, la cabeza lectora corre hacia la derecha, donde halla un 0, lo cambia por un x (lo marca), pasa al estado q_2.

$\textvisiblespace xx\overrightarrow{q_2}x\textvisiblespace$ Ahora en el estado q_2 la cabeza corriendo a la derecha se halla una x (un cero que marcamos en un paso anterior), se queda en el mismo estado, la cabeza corre a la derecha y no se cambia nada en la cinta.

$\textvisiblespace xxx\overrightarrow{q_2}\textvisiblespace$ Llegamos al final de la palabra, estamos en el estado q_2, cuando esto pasa se va al estado q_4, deja igual escrito el espacio vacío y la cabeza lectora empieza a correr ahora hacia la izquierda.

$\textvisiblespace xxx\overleftarrow{q_4}\textvisiblespace$ En el estado q_4, la cabeza lectora corre a la izquierda, se encuentra una x, no la altera, se queda en el mismo estado y la cabeza sigue corriendo a la izquierda.

$\textvisiblespace xx\overleftarrow{q_4}x\textvisiblespace$ Lo mismo del inciso anterior.

$\textvisiblespace x\overleftarrow{q_4}xx\textvisiblespace$ Lo mismo del inciso anterior.

$\textvisiblespace\overleftarrow{q_4}xxx\textvisiblespace$ Llegamos a la orilla izquierda de la cadena, se pasa al estado q_1 y la cabeza corre a la derecha.

$\textvisiblespace\overrightarrow{q_1}xxx\textvisiblespace$ Ahora en el estado q_1 la cabeza corriendo a la derecha se halla una x (un cero que marcamos en un paso anterior), se queda en el mismo estado, la cabeza corre a la derecha y no se cambia nada en la cinta.

$\textvisiblespace x\overrightarrow{q_1}xx\textvisiblespace$ Lo mismo del inciso anterior.

$\textvisiblespace xx\overrightarrow{q_1}x\textvisiblespace$ Lo mismo del inciso anterior.

$\textvisiblespace xxx\overrightarrow{q_1}\textvisiblespace$ Estamos en q_1, la cabeza corre a la derecha y se encuentra un espacio vacío, la máquina pasa al estado de aceptación y ya no importa que pase con la cabeza lectora y la cinta, ya llegó a su fin la máquina y la cadena fue aceptada.

$\sqcup xxx \sqcup r$ Fin, cadena aceptada.

Prueben con cadenas más grandes, pero ya se pueden dar una idea de como funciona la máquina, es un proceso recursivo que va dividiendo a mitades la cadena y checando que cada mitad tenga la longitud igual a una potencia de 2.

Un ejemplo con más de una cinta

Construir una máquina de Turing que cheque si una cadena binaria es un palíndromo.
Esquematiuzo los pasos a seguir de esta forma:

- Máquina de dos cintas, la entrada está en la primera cinta.
- Copiar la entrada de la primera cinta a la segunda.
- Regresa la cabeza lectora en la primera cinta al inicio
- Empieza a mover en sentidos contrarios para comparar, borrando cada carácter que coincida en la segunda cinta.

Las funciones de transición quedarían de la forma:

Cinta 1: $\delta(s, 0) \to (s, 0, \to)$
Cinta 2: $\delta(s, \sqcup) \to (s, 0, \to)$

Cinta 1: $\delta(s, 1) \to (s, 1, \to)$
Cinta 2: $\delta(s, \sqcup) \to (s, 1, \to)$

Cinta 1: $\delta(s, \sqcup) \to (q_1, \sqcup, \leftarrow)$
Cinta 2: $\delta(s, \sqcup) \to (q_1, \sqcup, -)$
Cinta 1: $\delta(q_1, 0) \to (q_0, 0, \leftarrow)$
Cinta 2: $\delta(q_1, \sqcup) \to (q_1, \sqcup, -)$

Cinta 1: $\delta(q_1, 1) \to (q_1, 1, \leftarrow)$
Cinta 2: $\delta(q_1, \sqcup) \to (q_1, \sqcup, -)$

Cinta 1: $\delta(q_1, \sqcup) \to (q_2, \sqcup, \to)$
Cinta 2: $\delta(q_1, \sqcup) \to (q_2, \sqcup, \leftarrow)$

Cinta 1: $\delta(q_2, 0) \to (q_2, 0, \to)$
Cinta 2: $\delta(q_2, 0) \to (q_2, \sqcup, \leftarrow)$

Cinta 1: $\delta(q_2, 1) \to (q_2, 1, \to)$
Cinta 2: $\delta(q_2, 1) \to (q_2, \sqcup, \leftarrow)$

Cinta 1: $\delta(q_2, 0) \to (r, -, -)$
Cinta 2: $\delta(q_2, 1) \to (r, -, -)$

Cinta 1: $\delta(q_2, 1) \to (r, -, -)$
Cinta 2: $\delta(q_2, 0) \to (r, -, -)$

Cinta 1: $\delta(q_2, \sqcup) \to (t, -, -)$
Cinta 2: $\delta(q_2, \sqcup) \to (r, -, -)$

Otro ejemplo, contando cadenas iguales

Construir una máquina de Turing que acepte dos cadenas iguales, sin separación distinguible del alfabeto $\{0, 1\}$. De nueva cuenta listo los pasos a seguir:

1. Checar que la longitud de la cadena sea múltiplo de 2 (primer barrido)
2. En el camino de regreso marcar el último carácter, identificando si es 0 o 1 y llevar la cuenta, para los demás caracteres no cambiar.
3. Al llegar al inicio de la cadena, o al primer carácter marcado reiniciar el proceso

4. Al repetir se llegará al punto en que todo está marcado, ahí nos aseguramos donde está la mitad de la cadena, ahora sí comparar

La tabla de transiciones:

	$\vdash$	0	1	X	Y	$\llcorner\!\lrcorner$
s	$(s, \vdash, \rightarrow)$	$(q_1, 0, \rightarrow)$	$(q_1, 1, \rightarrow)$			$(q_2, \llcorner\!\lrcorner, \leftarrow)$
q_1		$(s, 0, \rightarrow)$	$(s, 1, \rightarrow)$			$(r, -, -)$
q_2	$(t, -, -)$	$(q_3, X, \leftarrow)$	$(q_3, Y, \leftarrow)$	$(q_5, X, \rightarrow)$	$(q_5, Y, \rightarrow)$	
q_3	$(q_4, \vdash, \rightarrow)$	$(q_3, 0, \leftarrow)$	$(q_3, 1, \leftarrow)$	$(q_4, X, \rightarrow)$	$(q_4, Y, \rightarrow)$	
q_4		$(q_4, X, \rightarrow)$	$(q_4, Y, \rightarrow)$	$(q_2, X, \leftarrow)$	$(q_2, Y, \leftarrow)$	
q_5				$(q_6, 0, \leftarrow)$	$(q_7, 1, \leftarrow)$	
q_6	$(q_8, \vdash, \rightarrow)$	$(q_6, 0, \leftarrow)$	$(q_6, 1, \leftarrow)$	$(q_6, X, \leftarrow)$	$(q_6, Y, \leftarrow)$	$(q_8, \llcorner\!\lrcorner, \rightarrow)$
q_7	$(q_9, \vdash, \rightarrow)$	$(q_7, 0, \leftarrow)$	$(q_7, 1, \leftarrow)$	$(q_7, X, \leftarrow)$	$(q_7, Y, \leftarrow)$	$(q_9, \llcorner\!\lrcorner, \rightarrow)$
q_8				$(q_{10}, \llcorner\!\lrcorner, \rightarrow)$	$(r, -, -)$	
q_9				$(r, -, -)$	$(q_{10}, \llcorner\!\lrcorner, \rightarrow)$	
q_{10}		$(q_{11}, a, \rightarrow)$	$(q_{11}, b, \rightarrow)$	$(q_{10}, X, \rightarrow)$	$(q_{10}, Y, \rightarrow)$	
q_{11}	$(q_{11}, a, \rightarrow)$	$(q_{11}, b, \rightarrow)$	$(q_6, 0, \leftarrow)$	$(q_7, 1, \leftarrow)$	$(t, -.-)$	

Jerarquía aritmética

Hasta el momento hemos encontrado un problema paradigmático para establecer los limites de lo que es computable y lo que no, el problema de la detención (H.P. por sus siglas en inglés). A partir de él podemos catalogar el resto de problemas que nos encontremos si H.P. o su complemento, $\tilde{\text{H}}$.P. puede reducirse a ese otro problema.

Decimos que un problema B es Turing reducible a A, y lo escribimos $A \leq_T B$. De acuerdo a esa relación podemos clasificar los tipos de lenguaje como:

- $\Sigma_1^0 = \{$ lenguajes recursivamente enumerables$\}$
- $\Delta_1^0 = \{$ lenguajes recursivos$\}$
- $\Sigma_{n+1}^0 = \{$lenguajes recursivamente enumerables en algún $L \in \Sigma_n^0\}$
- $\Delta_{n+1}^0 = \{$lenguajes recursivos en algún $L \in \Delta_n^0\}$
- $\Pi_n^0 = \{$complemento de lenguajes en $\Sigma_n^0\}$

Esto se puede verbalizar en forma sintáctica como:

Un lenguaje es recursivamente enumerable si y sólo si existe una propiedad R decidible de pares de cadenas tal que $L = \{\alpha | \exists \beta$ tal que $R(\alpha, \beta)\}$.

De esta forma se puede escribir nuestro problema paradigmático como:

$$HP = \{\langle M\#\alpha\rangle | \exists x.M \text{ se detiene con } \alpha \text{ en } x \text{ pasos}\}$$

Y el problema de la pertenencia:

$$MP = \{\langle M\#\alpha\rangle | \exists x.M \text{ acepta a } \alpha \text{ en } x \text{ pasos}\}$$

Sabemos que ambos problemas son recursivamente enumerables, en este caso podemos identificarlo con sólo escribir el enunciado de la forma mencionada:

- Un lenguaje L está en Σ_n^0 si y sólo si existe una propiedad R $(n+1)$-aria[3] decidible tal que

$$L = \{\alpha | \exists \beta_1 . \forall \beta_2 ... \beta_n R(\alpha, \beta_1 ..., \beta_n)\}$$

- Un lenguaje L está en Π_n^0 si y sólo si existe una propiedad R $(n+1)$-aria decidible tal que

$$L = \{\alpha | \forall \beta_1 . \exists \beta_2 ... \beta_n R(\alpha, \beta_1 ..., \beta_n)\}$$

- Finalmente $\Delta_n^0 = \Sigma_n^0 \cap \Pi_n^0$

Recomendación de lectura referente a máquinas de Turing

Les dejo una recomendación para leer, no es un libro de ciencia ni de texto, menos divulgativo, es una novela policiaca.

Quizá conozcan una famosa serie de películas y novelas suecas, con *remake* estadounidense y una más reciente película que según dicen no se apega a la historia original, llamada Milennium: "Los hombres que no amaban a las mujeres", "La chica que soñaba con una cerilla y un bidón de gasolina", "La reina en el palacio de las corrientes de aire"(estas tres novelas son de Stieg Larsson, periodista y escritor sueco, también fueron hechas película en Suecia, y la primera tiene una version estadounidense, aunque grabada en Suecia, valen la pena), "Lo que no te mata te hace más fuerte", .El hombre que perseguía su sombraz "La chica que vivió dos veces"(estas últimas tres el autor original no las terminó, falleció incluso antes de que se publicaran las primeras tres, David Lagercrantz fue el escritor designado por la editorial para terminar la serie, de estas últimas tres hubo una supuesta adaptación no muy clara).

Son novelas policiacas que al parecer son muy populares en Suecia, a diferencia de las series y películas policiacas de Estados Unidos (y otros países que los siguen muy de cerca) los héroes no son detectives o policías honestos y honrados, o que de menos caigan bien (eso no suele suceder en la mayoría de países), son periodistas, y en esta serie en específico, un periodista y una *hacker*. Yo no he leído estas novelas, son muy extensas, quizá algún día. Lo interesante aquí es el autor de las últimas tres.

David Lagercrantz fue elegido por la editorial, en una maniobra plenamente comercial, para terminar la serie dado el éxito de un libro policiaco que escribió antes: .El enigma de Turing". Recabando algunos datos históricos, incluso con una excelente y muy bien explicada sección sobre el *Entscheidungsproblem*, Lagercrantz altera un poco la historia para contarla como una novela policiaca, donde Turing aparece en las anécdotas y las historias referentes a él (no es un personaje activo en la novela).

La novela, muy al gusto Sueco diría yo, es algo extensa, muy rica en descripciones, en generar el ambiente (en ocasiones se aleja demasiado de la historia para contar cosas aparentemente secundarias) pero al final justo salta la intriga, hay un poco de acción aunque no hay pistolas y solo un poco de sangre. Hay una

[3] Es decir, una propiedad es binaria si sólo es para un miembro en relación a la cadena mencionada: $R(\alpha, \beta)$. Sería terciaria si s relación a otros dos objetos: $R(\alpha, \beta, \gamma)$, etc.

adaptación cinematográfica, que no es mala, aunque un poco absurda en ciertas cosas, la novela es mucho mejor y va más allá de lo que cuenta la película.

Dirán: "Si bueno, mucha recomendación, ¿pero de donde la voy a sacar? Ha de ser costosa (seguramente es edición española), quién sabe si haya en las librerías cerca de casa y prefiero no salir a buscarla". Eso no es problema, si les interesa y tiene la posibilidad de leer en un lector electrónico, su celular o tableta (hay aplicaciones o en la misma configuración que les ayuda a hacer más tolerable la luz de la pantalla) o en la computadora (asegúrense de descansar la vista) yo se las presto muy amablemente, de cuates (no somos piratas porque no robamos barcos, yo ni nadar sé, no le robamos nada a nadie), el libro electrónico.

Descarguen el libro de: `https://www.dropbox.com/s/f010wkl7usewzet/Lagercrantz%2C%20David%20-%20El%20enigma%20Turing%20%5B36660%5D%20%28r1.2%29.epub?dl=0`. Esta en formato epub, hay varias aplicaciones libres para celular, tableta y computadora para leerlo.

Referencias

[1] Kozen, Dexter C. "Automata and Computability" Springer (1997)

[2] Sipser, Michael "Introduction to the Theory of Computation" 2a ed., Thomson Course Tecnology (2006)