

Maquina de Turing

Antes de empezar una aclaración: Este documento fue hecho a partir de los apuntes tomados en las clases de la profesora Silvina Ortega.

Se usa para gramáticas del tipo 1.

Existe una “cinta” en la que se graba la palabra a analizar y un carácter especial llamado fin o null, que obviamente indica el fin de la palabra.

La idea es recorrer la cinta sobrescribiendo los caracteres a medida que los leo para después controlar que todos hayan sido sobrescritos (obviamente existe una lógica para sobrescribir que me permite controlar que se cumplan las restricciones del lenguaje).

Para cambiar de estado se utiliza la siguiente terna:
(ϵ , opCinta, Mov)

donde:

- ϵ es el carácter que estoy leyendo de la cinta

-opCinta es el carácter que voy a escribir en el lugar del carácter leído. Esto indica que operación realizo en la cinta ya que si coloco el mismo carácter que había estoy leyendo, en cambio si coloco otro carácter estoy sobrescribiendo.

-Mov es hacia donde me voy a mover. Se indica R si me muevo hacia la derecha y L si me muevo hacia la izquierda.

La palabra es reconocida si se llega al estado final o de reconocimiento.

Algo a tener en cuenta es que en la maquina de turing el primer nodo se debe llamar ST y el ultimo se debe llamar STP (start y stop).

Lo voy a tratar de explicar con un ejemplo ya que es muy difícil de explicar en palabras.

Supongamos que queremos escribir la maquina de turing del siguiente lenguaje:

$L = \{a^n + b^n + c^n / n \geq 1\}$ (donde ^ es el símbolo que representa la potencia).

LA cinta será de la forma (tomando a μ como el carácter q indica el fin de la palabra):

a	...	b	...	c	...	μ
---	-----	---	-----	---	-----	-------

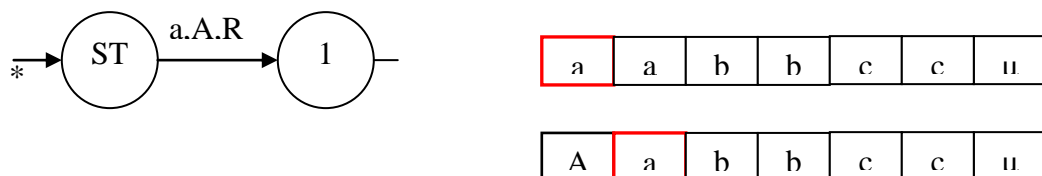
Algunas palabras serían por ejemplo:

a	a	b	b	c	c	μ
---	---	---	---	---	---	-------

a	a	a	b	b	b	c	c	c	μ
---	---	---	---	---	---	---	---	---	-------

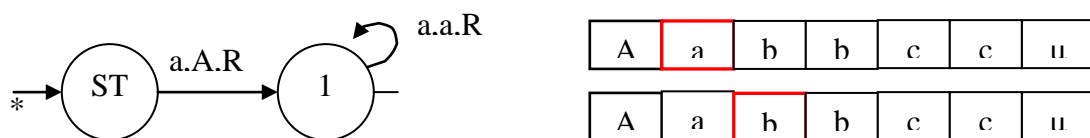
Para ir armando la maquina de turing se debe trabajar con la expresión general pero para facilitar el aprendizaje voy a armar la maquina con el primer ejemplo.

1) Voy a leer una a (ya que $n \geq 1$, una n va a haber seguro)



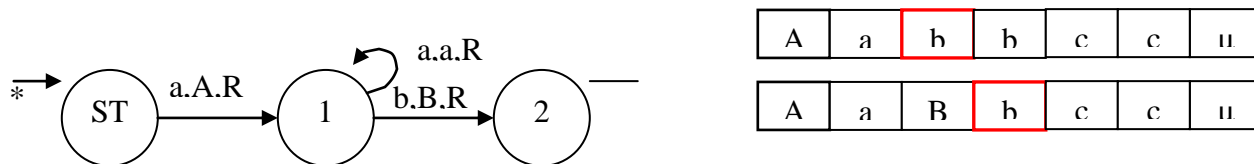
Nótese que replacé a por A para indicar que ya la leí y me moví a la derecha.

2) Debo tener en cuenta que puede haber más a y que debo leer de a un conjunto por vez, es decir que cuando leo una a debo leer una b y una c para luego leer la siguiente a:

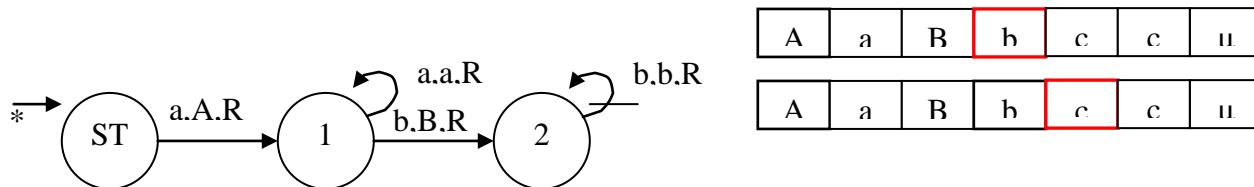


Entonces cuando lea otra a la voy a reemplazar por una a, es decir, no la voy a cambiar.

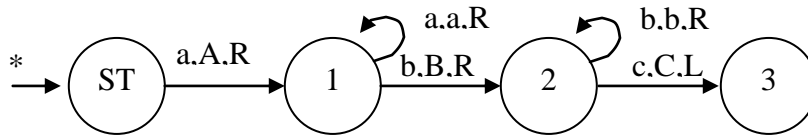
3) En cuanto encuentre la primera b debo reemplazarla por una B.



4) Igual que 2 pero con b:



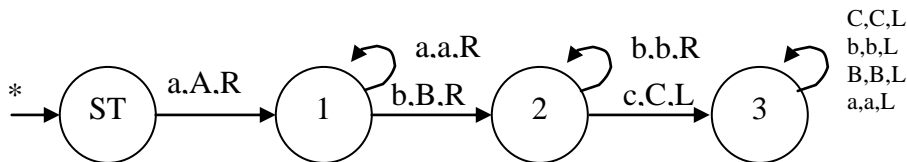
5) Cuando aparezca la primera c lo que debo hacer es reemplazarla por C como en los casos anteriores pero esta vez me voy a mover para la izquierda (como ya conseguí la c que necesitaba para completar el conjunto, ahora debo volver para atrás para analizar el próximo conjunto de abc).



A	a	B	h	c	c	u
---	---	---	---	---	---	---

A	a	B	h	C	c	u
---	---	---	---	---	---	---

6) Debo volver hasta la última a que encuentre, al hacer eso me puedo llegar a encontrar con C, b, B o a así que:



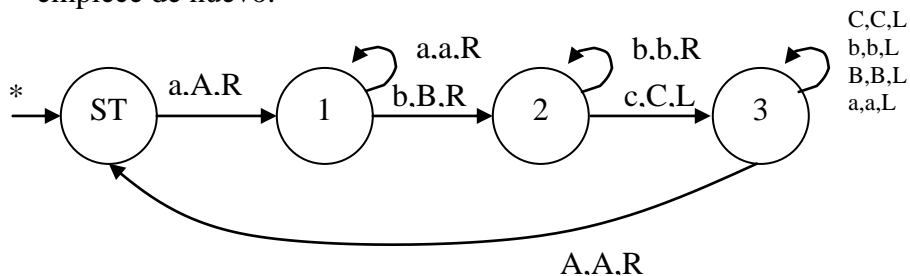
A	a	B	h	C	c	u
---	---	---	---	---	---	---

A	a	B	h	C	c	u
---	---	---	---	---	---	---

A	a	B	h	C	c	u
---	---	---	---	---	---	---

A	a	B	h	C	c	u
---	---	---	---	---	---	---

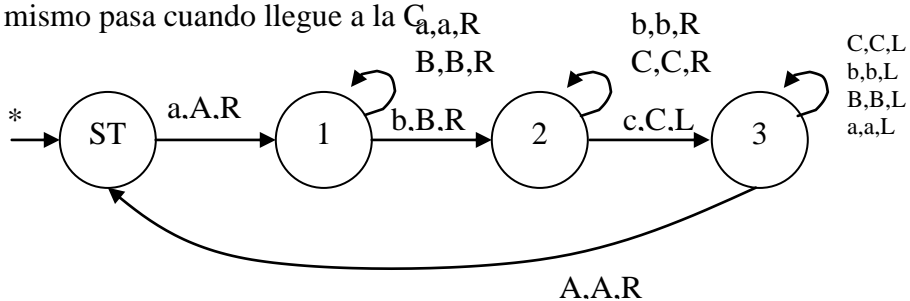
7) Cuando llegue al ultimo a, hago un “artilugio” para volver al nodo ST y que el ciclo empiece de nuevo.



A	a	B	h	C	c	u
---	---	---	---	---	---	---

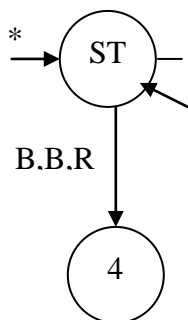
A	a	B	h	C	c	u
---	---	---	---	---	---	---

8) Vuelvo a hacer los pasos anteriores, pero ocurre algo: en la cinta ahora hay una B ,entonces, debo AGREGAR en el bucle, leer también la B y reemplazarla por otra B. Lo mismo pasa cuando llegue a la C_{a,a,R}



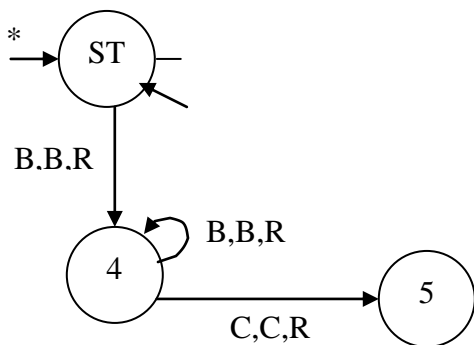
A	a	B	b	C	c	␣
A	A	R	b	C	c	␣
A	A	B	h	C	c	␣
A	A	B	B	C	c	␣
A	A	B	B	C	c	␣
A	A	B	B	C	C	␣
A	A	B	R	C	C	␣
A	A	R	B	C	C	␣
A	A	B	B	C	C	␣

8) Ahora tenemos toda la cinta leída, ahora viene la parte en que reconocemos si la palabra pertenece o no a el lenguaje(no voy a dibujar los nodos 1, 2 y 3 pero deben estar).



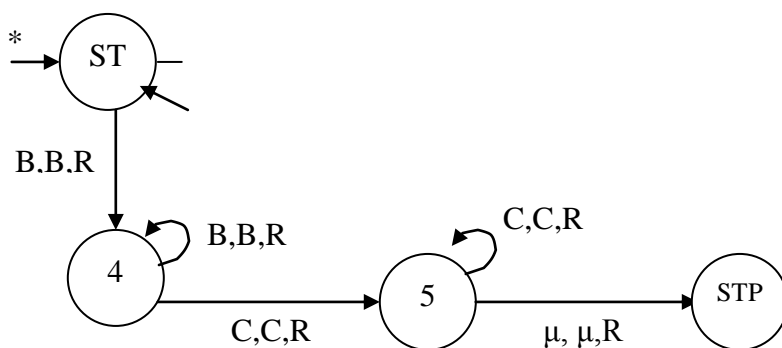
A	A	B	B	C	C	␣
A	A	R	B	C	C	␣

9) Puede haber mas de una B así que repetimos hasta q aparezca una C



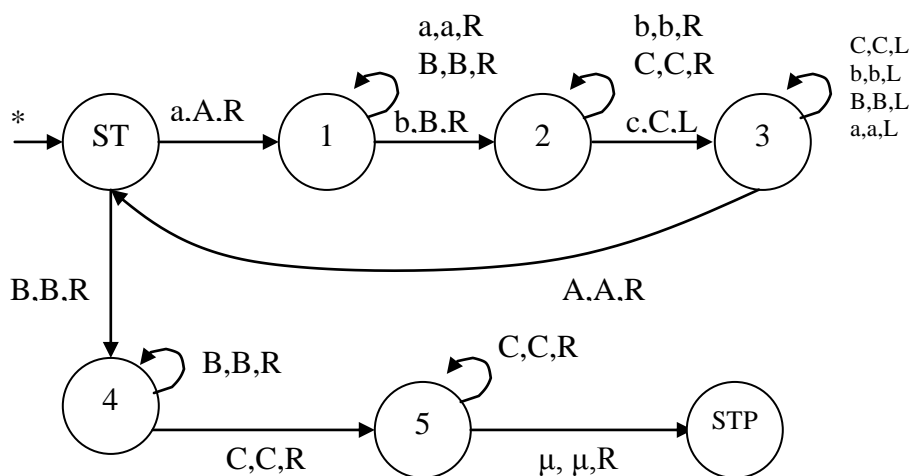
A	A	B	B	C	C	␣
A	A	B	B	C	C	␣
A	A	B	B	C	C	␣

10) Puede haber más de una C así que debo repetir hasta que aparezca el fin de cadena.



A	A	B	B	C	C	␣
A	A	B	B	C	C	␣
A	A	B	B	C	C	␣

Entonces la Maquina quedara así:



Otro Ejemplo:

$$L = \{a^{2n} + b^n + c^n \mid n \geq 1\}$$

