

Construcción de un analizador léxico de una ER, continuación

Que para coadyuvar en la calificación del primer parcial

Alumno

Instituto Politécnico Nacional

Agosto de 2018

Trabajo: transformar un autómata AFN en analizador léxico de una ER

Objetivo

Utilizar la construcción de Thompson , los algoritmos de subconjuntos y de minimización para convertir el autómata AFN en un autómata AFD mínimo.

Las operación de cerradura.

DEFINICIÓN (Cerradura- ϵ .)

La cerradura- $\epsilon(\{S\})$ es el conjunto de todos los estados que reciben una transición con el caracter vacio ϵ , incluido el conjunto de estados $\{S\}$ de donde salen las transiciones ϵ .

La operación de movimiento

DEFINICIÓN (Movimiento(Estado, Símbolo).)

*Es el movimiento de un estado marcado como **Estado** con cada símbolo del alfabeto Σ hacia otro estado. El movimiento se puede expresar también como **mov(estado, símbolo)**.*

Autómata AFN obtenido de la construcción de Thompson.

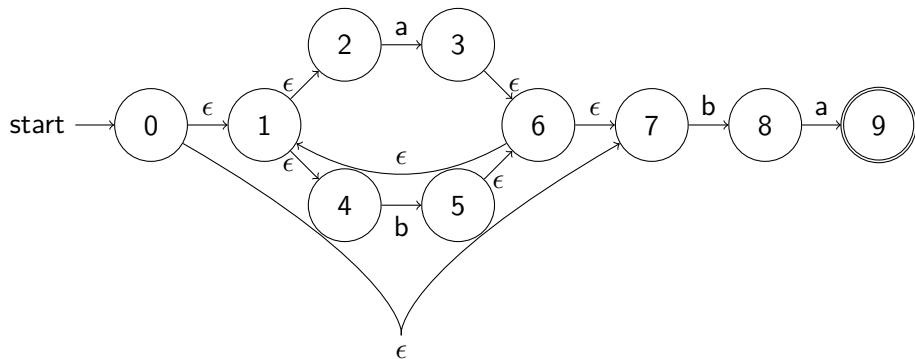


Figura: Autómata AFN obtenido de la expresión regular **Itálica** $\{(a|b)^*ba\}$.

Autómata AFD no mínimo.

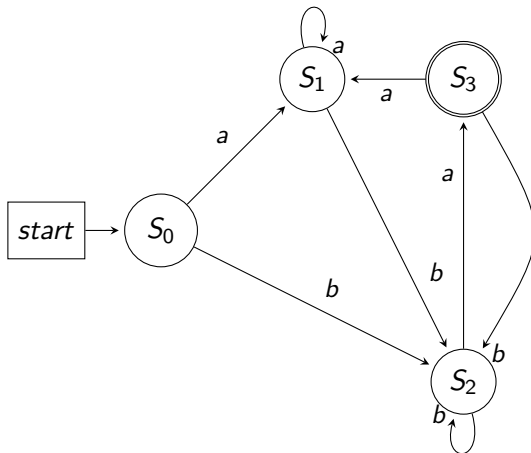


Figura: AFD obtenido de la expresión regular **Itálica** $\{(a|b)^*ba\}$.

Autómata AFD mínimo.

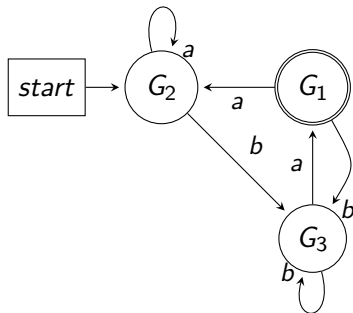
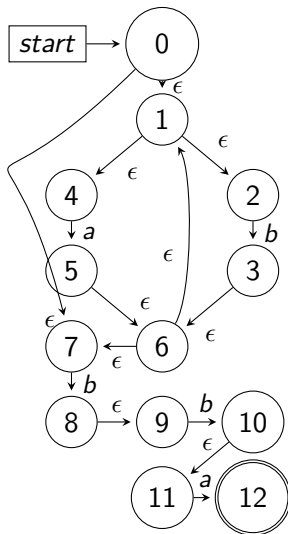


Figura: AFD mínimo obtenido de la expresión regular **Itálica** $\{(a|b)^*ba\}$.

AFN obtenido de los patrones de la construcción de Thompson de la expresión regular **Itálica** $\{(a|b)^*bba\}$.



Autómata AFD no mínimo.

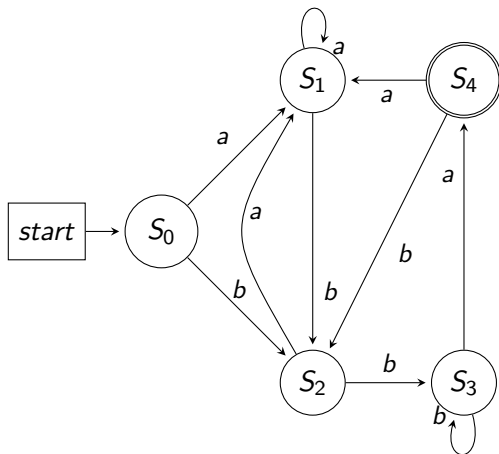


Figura: AFD obtenido de la expresión regular **Itálica** $\{(a|b)^*bba\}$.

Construcción del reconocedor

Algorithm .1: SiguienteCaracter(), donde p es una lista enlazada simple, que contiene la palabra a analizar.

```
1 TIPE c:char ;
2 TIPE  $\uparrow p$ :Lista ;
3 while  $p \uparrow .sig \neq NULL$  do
4    $c := p \uparrow .dato$  ;
5    $\uparrow p := p \uparrow .sig$  ;
6   return c ;
```

Construcción del reconocedor

Algorithm .2: Instrucciones para un reconocedor

```
1  $car \leftarrow \text{SiguieteCaracter}();$   
2  $Estados \leftarrow s_0;$   
3 while  $car \neq EOF \wedge Estados \neq s_e$  do  
4    $Estados := \delta[Estados, car];$   
5    $car := \text{SiguieteCaracter}();$   
6 if  $Estados == s_a$  then  
7    $EsEnTec(Aceptado);$   
8 else  
9    $EsEnTec(Error);$ 
```
