INSTITUTO POLITÉCNICO NACIONAL

Escuela Superior de Computación

Materia:

Teoría Computacional

Lista de ejercicios

Profesor:

Diaz Santiago Ricardo Felipe

Grupo:

1CM8

Alumno:

- Castro cruces Jorge Eduardo _______________
- Guzmán Gutierrez Manuel _______________

Capitulo 3.

3.1.1 Escriba expresiones regulares para los siguientes lenguajes:

a) El conjunto de cadenas del alfabeto {a,b,c} que contienen al menos una a y al menos una

Sea L = {abc} - L⁺ = {abc, aabbcc, aaabbbccc, ...}

$a^+b^+c^*$

3.1.2. Escriba expresiones regulares para los siguientes lenguajes

a) El conjunto de todas las cadenas formadas por ceros y unos tales que cada pareja de 0s adyacentes aparece antes que cualquier pareja de 1s adyacentes

C = {00 11 00 11}

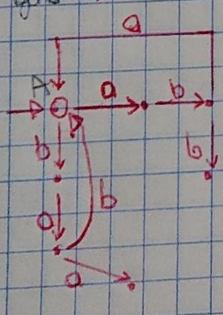
$L = (00 + 11)^+$

3.1.4 Proporcione las descripciones informales de los lenguajes correspondientes a las siguientes expresiones regulares

a) $(1+\varepsilon)(00^*1)^*0^*$: El conjunto de cadenas formadas necesariamente con un 1, o nada, seguido de n veces cero (incluido la cadena vacía), seguido de un uno, eso n veces, en la que se encuentra también la cadena vacía, seguida de n veces 0, en las que también puede estar la cadena vacía.

b) $(0^*1^*)^*000(0+1)^*$. El conjunto de cadenas de n veces 0 junto a n veces 1, es (dentro de lo que en ambos puede estar la cadena vacía) eso n, dentro de lo que puede estar la cadena vacía seguido por 3 ceros consecutivos, unido a la unión de un cero y uno n veces, dentro de lo que puede aparecer la cadena vacía.

3.2.1 Construir una gramática regular para el lenguaje regular aceptado por el autómata finito de la Figura 3.4



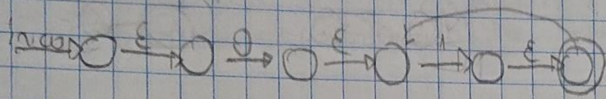
S → aB | bA | 1

A → a | aB

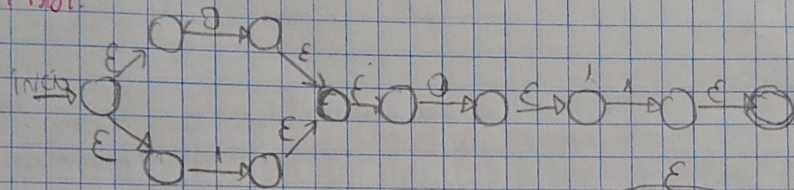
B → bA | aS

3.2.4 Convierta las siguientes expresiones regulares en un AFN con transiciones-ε

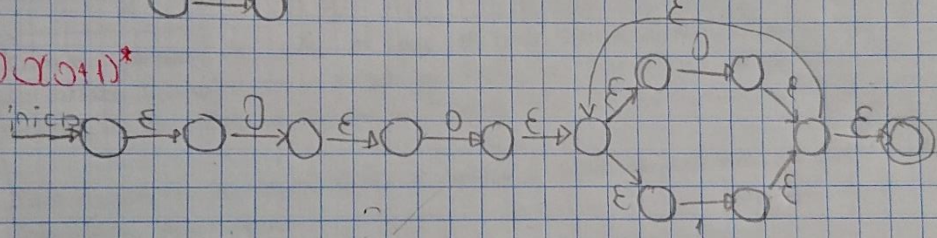
a) 01^*.



b) $(0+1)01$.

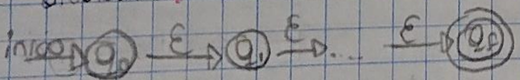


c) $00(0+1)^*$



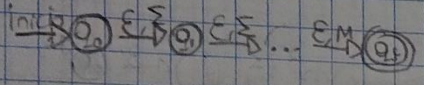
3.2.6 Sea $A=(Q,\Sigma,\delta,q_0,\{q_f\})$ un AFN-ε tal que no existen transiciones a q_0 ni tampoco transiciones que salgan de q_f. Describa el lenguaje aceptado por cada una de las siguientes modificaciones de A, en función de $L=L(A)$:

a) El autómata construido a partir de A añadiendo una transición-ε de q_0 a todos los estados alcanzables.



El autómata aceptará la cadena ε un n número de veces, dependiendo la cantidad de estados alcanzables por q_0, hasta llegar al estado final q_f

b) El autómata construido a partir de A añadiendo una transición-ε de q_0 a todos los estados alcanzables desde q_0 (a lo largo de un camino cuyas etiquetas pueden incluir símbolos de Σ así como ε).



El autómata aceptará todas las cadenas vacías, o cualquier símbolo que se encuentre dentro de Σ, un n número de veces, dependiendo de la cantidad de estados alcanzables por q_0 hasta llegar al estado final (q_f).

3.2.7 Existen algunas simplificaciones para las construcciones del teorema 3.7, en el que convertimos una expresión regular en un AFN-ε. He aquí tres de ellas:

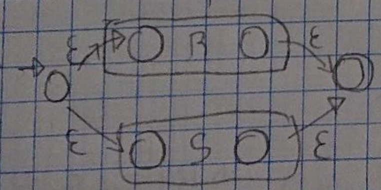
1. Para el operador de unión, en lugar de crear nuevos estados inicial y de aceptación, se combinan los dos estados iniciales en un estado con todas las transiciones de ambos estados. Del mismo modo, se combinan los dos estados de aceptación, teniendo todas las transacciones que ir al estado combinado.

2. Para el operador de concentración se combina el estado de aceptación del primer autómata con el estado inicial del segundo.

3. Para el operador de clausura, simplemente se añaden transacciones-ε del estado de aceptación al estado inicial y viceversa.

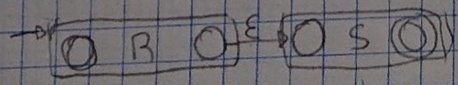
Cada una de estas simplificaciones, por sí misma, genera una construcción correcta; es decir, el AFN-ε resultante para cualquier expresión regular acepta el lenguaje de la expresión. ¿Qué subconjuntos en los cambios (1), (2) y (3) pueden aplicarse a la vez, dando un lugar a un autómata correcto para toda expresión regular?

1.



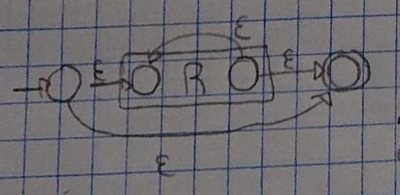
La expresión es de la forma R+S para dos expresiones R+S más pequeñas. Es decir, partiendo del nuevo estado inicial se puede llegar al estado inicial del autómata correspondiente a R o al correspondiente a S.

2.



La expresión es de la forma RS para expresiones RS más pequeñas.

3.



La expresión es de la forma R^* para una expresión R más pequeña. El autómata permite ir directamente del estado inicial al de aceptación y volver a R.

3.2.8 Proporcione un algoritmo que parta de un AFD A y calcule el número de cadenas de longitud n (para cierto n dado, no relacionado con el número de estados de A) aceptadas por A. El algoritmo deberia ser polinómico tanto respecto a n como al número de estados de A. Utilice la técnica sugerida por la construcción del teorema 3.4.

Si $L = L(A)$, para algún AFD A, entonces existe una expresión R tal que $L = L(R)$
Supongamos que los estados de A son $\{1, 2, \ldots, n\}$ para algún entero n.

El caso base es para $k = 0$. 1- Un arco desde el nodo (estado) i hasta nodo j, 2- Un camino de longitud 0 que consta sólo de algún nodo i.
Si $i \neq j$, entonces sólo es posible el caso (1).
a) Si no existe tal símbolo a, entonces $R_{ij}^{(0)} = \emptyset$
b) Si existe solamente un símbolo a, entonces $R_{ij}^{(0)} = a$.
c) Si existen símbolos $a_1, a_2, \ldots, a_k$ que etiquetan arcos desde el estado i hasta el estado j, entonces $R_{ij}^{(0)} = a_1 + a_2 + \ldots + a_k$.
Sin embargo, si $i = j$ entonces los caminos válidos son el camino de longitud 0 y todos los bucles desde i a sí mismo.
Suponga que existe un camino desde el estado i hasta el estado j que no pasa por ningún estado mayor que k.

Si combinamos las expresiones para los caminos de los tipos anteriores, tenemos la expresión

$$R_{ij}^{(k)} = R_{ij}^{(k-1)} + R_{ik}^{(k-1)}(R_{kk}^{(k-1)})^* R_{kj}^{(k-1)}$$ para las etiquetas de todos los caminos desde el estado i al estado j que no pasa por ningún estado mayor que k.

3.4.1 Verifique las siguientes identidades que utilizan expresiones regulares

a) $R + S = S + R$

Sea x cualquier elemento, entonces $x \in (R \cup S)$

$$x \in (R \cup S) \Rightarrow x \in R \lor x \in S$$
$$\Rightarrow x \in S \lor x \in R$$
$$\Rightarrow x \in (S \cup R)$$

$$\forall x [x \in R \cup S \Leftrightarrow x \in S \cup R] \Rightarrow \therefore R \cup S = S \cup R \Rightarrow R + S = S + R$$

b) $(R + S) + T = R + (S + T)$

Sea x cualquier elemento arbitrario,

$$x \in R \cup (S \cup R) \Leftrightarrow x \in R \lor [x \in (S \cup T)]$$
$$\Leftrightarrow x \in R \lor (x \in S \lor x \in T)]$$
$$\Leftrightarrow (x \in R \lor x \in S) \lor x \in T$$
$$\Leftrightarrow (x \in (R \cup S)) \lor x \in T$$

b) $(RS+R)^*R = R(SR+R)^*$

$((R \cap S) \cup R)^* \cap R = R \cap ((S \cap R) \cup R)^*$

$(R)^* \cap R = R \cap (R)^*$

$R^+ = R^+$

$\therefore$ La expresión $(RS+R)^*R = R(SR+R)^*$ son equivalentes

3.4.3 En el ejemplo 3.6 hemos desarrollado la expresión regular:

$$(0+1)^*1(0+1)+(0+1)^*1(0+1)(0+1)$$

Utilizando las leyes distributivas desarrolle dos expresiones equivalentes diferentes y más simples.

$$(0 \cup 1)^* \cap 1 \cap (0 \cup 1) \cup (0 \cup 1)^* \cap 1 \cap (0 \cup 1) \cap (0 \cup 1)$$

$$(0 \cup 1)^* \cap 1 \cup (0 \cup 1)^* \cap 1 \cap (0 \cup 1)$$

$$(0 \cup 1)^* \cap 1 \cup (0 \cup 1)^* \cap 1 = 1^* \cup 1^* = 1^*$$

Como se llegó a la máxima simplificación posible, implícitamente ya vemos distintas expresiones dentro del proceso de simplificación

$x \in (R \cup S) \cup T$

$\forall x [x \in R \cup (S \cup T) \Leftrightarrow x \in (R \cup S) \cup T]$

$\therefore R \cup (S \cup T) = (R \cup S) \cup T \Rightarrow R + (S + T) = (R + S) + T$

c) $(RS)T = R(ST)$

Sea x un elemento arbitrario

$x \in (R \cap S) \cap T \Leftrightarrow [x \in (R \cap S)] \wedge x \in T$
$\Leftrightarrow x \in R \wedge (x \in S \vee x \in T)$
$\Leftrightarrow x \in R \wedge (x \in S \cap T)$
$\Leftrightarrow x \in R \cap (S \cap T)$

$\therefore \forall x [x \in (R \cap S) \cap T \Leftrightarrow x \in R \cap (S \cap T)]$

$\therefore (R \cap S) \cap T = R \cap (S \cap T) \Rightarrow (RS)T = R(ST)$

d) $R(S + T) = RS + RT$

Sea x un elemento arbitrario

$x \in R \cap (S \cup T) \Leftrightarrow x \in R \wedge [x \in (S \cup T)]$
$\Leftrightarrow x \in R \wedge (x \in S \vee x \in T)$
$\Leftrightarrow (x \in R \wedge x \in S) \vee (x \in R \wedge x \in T)$
$\Leftrightarrow x \in (R \cap S) \vee (R \cap T)$

$\forall x [x \in R \cap (S \cup T) \Leftrightarrow x \in (R \cap S) \cup (R \cap T)]$

$\therefore R \cap (S \cup T) = (R \cap S) \cup (R \cap T) \Rightarrow R(S + T) = RS + RT$

3.4.2 Demuestre si cada uno de las siguientes proposiciones acerca de expresiones regulares es verdadera o falsa.

a) $(R + S)^* = R^* + S^*$ $(\{R\} \cup \{S\})^* = \{R\}^* + \{S\}^*$

$\{R, S\}^* = \{R\}^* + \{S\}^*$ $\{R, S\}^* = \{R^*, S^*\}$

$\Rightarrow \{\varepsilon, R, S, RR, SS, RS, SR, \ldots\} = \{\varepsilon, R, S, RR, SS, RS, SR\}$

$\therefore$ Las expresiones $(R + S)^* = R^* + S^*$ son iguales