



Universidad Autónoma de Chiapas
Facultad de Contaduría y Administración, Campus I
Licenciatura en Ingeniería y Desarrollo de Tecnologías de Software



Materia: Compiladores

Actividad 1.

Docente: Dr. Gutiérrez Alfaro Luis.

Alumno: Georgina Andrea Méndez Hernández.

Matricula: A220001.

Tuxtla Gutiérrez Chiapas.

09 de agosto 2023.

Las expresiones regulares Reg Ex, no son un lenguaje de programación, sino una serie de símbolos (expresiones o patrones) que nos permitirán definir patrones de búsqueda en cadenas de texto.

Por lo tanto las expresiones regulares, no son nada más y nada menos que un lenguaje para manipular/realizar búsquedas sobre archivos de texto o simplemente sobre texto sin la necesidad de que estos estén contenidos en el interior de un fichero.

Operadores regulares

Existen 3 operadores regulares Unión, concatenación, y cerradura. Si L y M son dos lenguajes, su unión se denota por $L \cup M$ e.g

$L = \{001, 10, 111\}$, $M = \{\epsilon, 001\}$, la unión será $L \cup M = \{\epsilon, 10, 001, 111\}$.

La concatenación de lenguaje se denota como LM o $L.M$ e.g

$L = \{001, 10, 111\}$ $M = \{\epsilon, 001\}$ entonces la concatenación será $LM = \{001, 10, 111, 001001, 10001, 111001\}$.

Operadores de expresiones regulares

a) Cerradura Kleene, se representa por el símbolo $*$. Este símbolo indica la repetición de cero o más veces el símbolo que le antecede.

b) Cerradura positiva, se representa por el símbolo $+$, indica la repetición de uno o más veces el símbolo que le antecede.

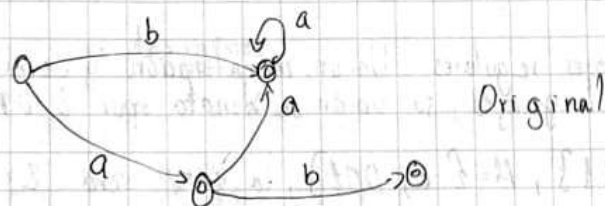
Por ejemplo la expresión regular a^+ indica repetir uno o

el símbolo a , la expresión a^+ es un conjunto de cadenas con el símbolo a que contiene los elementos $a^+ = \{a, aa, aaa, aaaa\}$

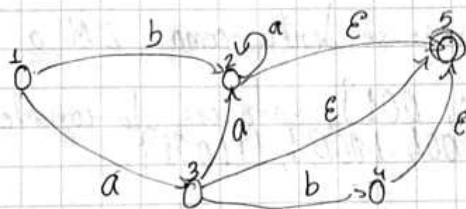
c) La concatenación por el símbolo \cdot

d) La disyunción por el símbolo $|$

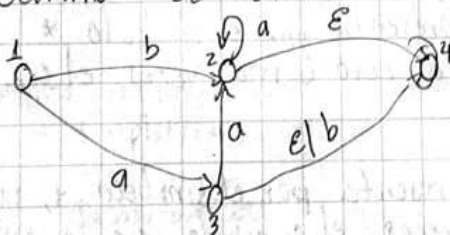
Convertir AFD a ER por eliminación de estados



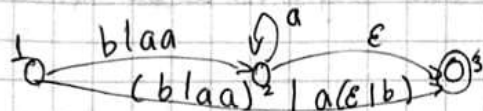
Se agrega un nuevo estado final



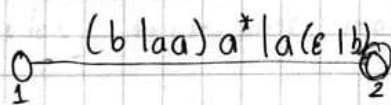
Se elimina el estado 4



Se elimina el estado 3



Se elimina finalmente el estado 2 para obtener la expresión regular



Se reducen términos

$$(b|a)^*|a(\epsilon|b)$$

$$= b^*|a^*|a(\epsilon|b)$$

$$= b^*|a^*|a|ab$$

$$= b^*|a^*|ab$$

$$= b^*|a^*|a|ab$$

$$= b^*|a^*|a(\epsilon|b)$$

Leyes algebraicas de las expresiones regulares

Son reglas y propiedades que describen cómo se pueden manipular y simplificar las expresiones regulares.

Leyes conmutativas

$$-(A+B \hat{=} B+A)$$

- $\neg (AB=BA)$ (solo si los lenguajes A y B son independientes del contexto, no en el caso general de las expresiones regulares).

Leyes Asociativas

- $\neg (A + (B + C) = (A + B) + C)$

- $\neg (A(BC) = (AB)C)$

Leyes distributivas

- $\neg (A(B+C) = AB+AC)$

- $\neg (A(BC) = (AB)C)$

- $\neg ((A+B)C = AC+BC)$

Patrón

Es un string que describe una serie de reglas a cumplir por otro string del que, en principio, no sabemos nada. Los patrones consumen muchos recursos de cálculo, con lo que es recomendable usarlos cuando es necesario.

Está compuesto por

- dos delimitadores, uno de inicio y otro de fin, normalmente con el carácter `/`

- uno o varios meta caracteres, que representarán el tipo de reglas que los caracteres del string de entrada deben cumplir.

Token
Secuencia de caracteres con significado sintáctico propio.

Lexema
Secuencia de caracteres cuya estructura se corresponde con el patrón de un token.

Ejemplos expresiones regulares

[^abc] Cualquier carácter menos : "a", "b", "c"

[a-z] Cualquier minúscula entre la "a" y la "z"

[0-9] Cualquier número entre el 0 y el 9

$^1d\{3\}$ 901, 304, 568 dígitos de 3 números