



UNIVERSIDAD AUTÓNOMA DE CHIAPAS
FACULTAD DE CONTADURÍA Y ADMINISTRACIÓN, CAMPUS I
Licenciatura en Ingeniería en Desarrollo y Tecnologías de
Software

DOCENTE: DR. GUTIERREZ ALFARO LUIS

MATERIA: COMPILADORES

ACTIVIDAD 1:

INVESTIGACIÓN Y EJEMPLOS

ALUMNA: DIANA LAURA VELASCO TORRES

MATRICULA: A210390

SEMESTRE: 6°

GRUPO: M

FECHA: 27 DE ENERO DEL 2024

LUGAR: TUXTLA GUTIÉRREZ, CHIAPAS

I.- Explicar los tipos de operadores de expresiones regulares.

Comúnmente existen tres operadores de las expresiones regulares: Unión, concatenación y cerradura.

1. Unión: Este operador permite combinar dos expresiones regulares en una sola que coincide con las cadenas que coinciden con cualquiera de las dos expresiones originales. (*CAPTUL1.PDF*, 2017)
2. Concatenación: Este operador une dos expresiones regulares en una sola que coincide con las cadenas que coinciden con la primera expresión seguida inmediatamente por una cadena que coincide con la segunda expresión. (*CAPTUL1.PDF*, 2017)
3. Clausura (o estrella de Kleene): Este operador se aplica a una sola expresión regular y produce una nueva expresión que coincide con cero o más repeticiones de la expresión original. (*CAPTUL1.PDF*, 2017)

Operadores comunes

Para definir patrones de coincidencia, puede utilizar estos operadores comunes:

Operador	Descripción	Ejemplo	Devuelve
^	Coincide con el principio de una cadena	^abc	abc, abcdef..., abc123
\$	Coincide con el final de una cadena	abc\$	mi:abc, 123abc, theabc
.	Coincide con cualquier carácter como comodín	a.c	abc, asc, a123c
 	Un carácter O	abc xyz	abc o xyz
(...)	Captura los valores entre paréntesis.	(a)b(c)	a y c
[...]	Coincide con todo lo que esté entre corchetes	[abc]	a, b, o c

[a-z]	Coincide con los caracteres en minúscula entre a y z	[b-z]	bc, mente, xyz
[0-9]	Coincide con cualquier valor numérico entre 0 y 9.	[0-3]	3201
{x}	El número exacto de veces que debe coincidir	(abc){2}	abcabc
{x,}	El número mínimo de veces que debe coincidir	(abc){2,}	abcabcabc
*	Coincide con cualquier cosa en lugar de *, o una coincidencia "codiciosa".	ab*c	abc, abbcc, abcdc
+	Coincide con el carácter anterior al + una o más veces	a+c	ac, aac, aaac
?	Coincide con el carácter anterior a ? cero o una vez, o una coincidencia "no codiciosa".	ab?c	ac, abc
/	Escapa el carácter después de /, O crea una secuencia de escape	a/bc	a c, con el espacio correspondiente a /b

Coincide con el inicio o el final de la cadena (^ y \$)

Para hacer coincidir patrones al principio o al final de la cadena, utilice los operadores ^ y \$, respectivamente. Por ejemplo:

Ejemplo	Partidos
^El	Cualquier cadena que empiece por El
de desesperación\$	Cualquier cadena que termine con of despair

^abc\$	Una cadena que empieza y termina con abc-una coincidencia exacta
---------------	--

Caracteres coincidentes (*, +, y ?)

Ejemplo	Partidos
ab*	Una cadena que contiene a, seguida de cero o más bs-ac, abc, o abbc
ab+	Una cadena que contiene a, seguido de uno o más bs-abc o abbc, pero no ac
¿Ab?	Una cadena que contiene a, seguido de cero o uno bs-ac o abc, pero no abc
a?b+\$	Una cadena que termina con uno o más bs, con o sin un a precedente ; por ejemplo, ab, abb, b, o bb, pero no aab o aabb

Para hacer coincidir patrones basados en un carácter específico, siga el carácter con el operador *, +, o ?. Estos operadores indican el número de veces que debe aparecer el carácter para obtener una coincidencia: cero o más, uno o más, o uno o cero, respectivamente. (*Operadores De Expresiones Regulares*, n.d.)

Para buscar un patrón basado en la frecuencia de aparición de un único carácter, escriba a continuación el número o el intervalo de casos, entre llaves ({...}). (*Operadores De Expresiones Regulares*, n.d.)

Ejemplo	Partidos
ab{2}	Una cadena que contiene a, seguida de exactamente 2 bs-abb
ab{2,}	Cadena que contiene a, seguida de al menos 2 bs-abb, abbbb, etc.
ab{3,5}	Una cadena que contiene a, seguida de tres a cinco bs-abbb, abbbb, o abbbbbb

Coincidencia de uno de varios patrones (|)

Coincide con cualquier carácter (.)

“Para representar cualquier carácter en un patrón a comparar, utilice el operador comodín `.`“(Operadores De Expresiones Regulares, n.d.)

Ejemplo	Partidos
<code>a.[0-9]</code>	Una cadena que contiene un, seguido de cualquier carácter y un dígito
<code>^. {3}\$</code>	Cualquier cadena de exactamente tres caracteres

Coincidencia de posición de caracteres (`[...]`)

Para hacer coincidir un patrón basado en la posición de un carácter, utilice paréntesis (`[...]`). (Operadores De Expresiones Regulares, n.d.)

Por ejemplo:

EJEMPLO	PARTIDOS
<code>[AB]</code>	Una cadena que contiene a o b; equivalente a <code>a b</code>
<code>[A-D]</code>	Cadena que contiene una minúscula a, b, c, o d; equivalente a <code>a b c d</code> o <code>[abcd]</code> .
<code>^[A-ZA-Z]</code>	Una cadena que empieza por cualquier letra, independientemente de mayúsculas y minúsculas
<code>[0-9]%</code>	Una cadena que contiene cualquier dígito seguido de un signo de porcentaje
<code>,[A-ZA-Z0-9]\$</code>	Una cadena que termina con una coma seguida de cualquier carácter

Coincidencia de caracteres no deseados (`[^...]`)

Para que coincida con un patrón que **no** contenga caracteres, inicie la secuencia con un operador `^` y enciérrela entre corchetes. Por ejemplo, `%[^a-zA-z]%` coincide

con una cadena con cualquier carácter no alfabético entre dos signos de porcentaje.
(*Operadores De Expresiones Regulares*, n.d.)

II.- Explicar el proceso de conversión de DFA a expresiones regulares.

El proceso de transformar un autómata finito determinista (DFA) en una representación regular se puede simplificar eliminando estados y cambiando aristas. Aquí hay una descripción general del proceso:

Comprenda los conceptos básicos de DFA y las expresiones regulares: DFA es un modelo matemático para reconocer y aceptar lenguajes regulares. Consiste en un conjunto finito de estados, un alfabeto de símbolos de entrada, una función de transición que asigna cada estado y símbolo de entrada a un nuevo estado, un estado inicial y un conjunto de estados de aceptación. (Juancar, 2013)

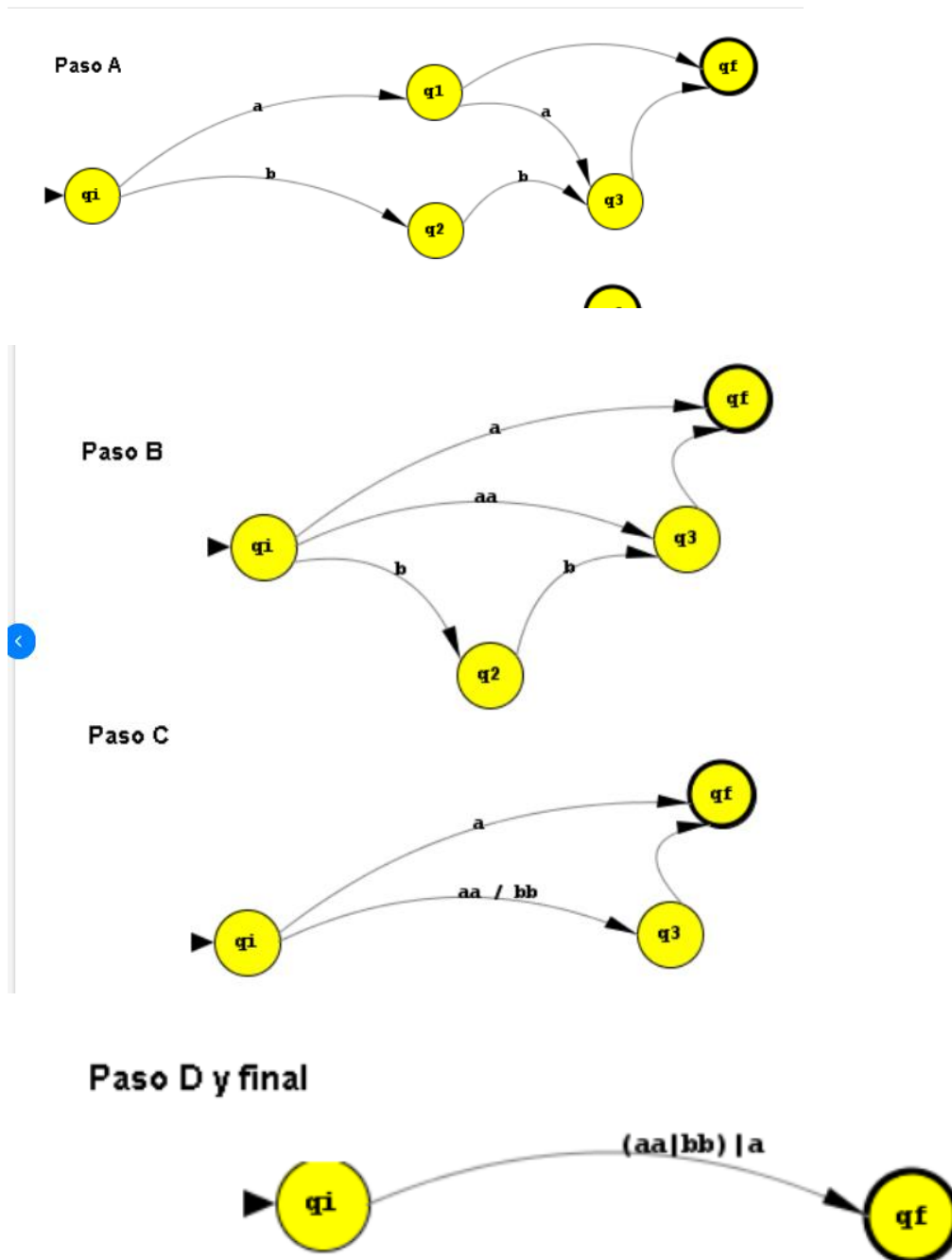
Definir estados no esenciales: los estados no esenciales son estados que no contribuyen a la aceptación de ninguna cadena de idioma. Al eliminar estos modos, simplificamos el DFA y reducimos su complejidad. (Juancar, 2013)

Minimización de estados: la minimización de estados implica dividir un conjunto de estados en clases de equivalencia en función de su diferenciación. (Juancar, 2013)

Eliminar condiciones no esenciales: una vez identificadas las condiciones no esenciales, se pueden eliminar de DFA. (Juancar, 2013)

Cambiar límites: los límites que conectan estos estados deben cambiarse para preservar el idioma reconocido por DFA. Esta modificación incluye la redirección de borde para evitar la eliminación del estado. (Juancar, 2013)

EJEMPLO:



23Proceso de conversión de AFD a expresión regular

(23Proceso de conversión de AFD a expresión regular | Download Scientific Diagram,n.d.)

III.- Explicar leyes algebraicas de expresiones regulares.

Existen un conjunto de leyes algebraicas que se pueden utilizar para las expresiones regulares:

- **Ley conmutativa para la unión:** $L + M = M + L$
- **Ley asociativa para la unión:** $(L + M) + N = L + (M + N)$
- **Ley asociativa para la concatenación:** $(LM)N = L(MN)$ (La concatenación no es conmutativa, es decir $LM \neq ML$) (*Expresiones Regulares*, 2015)
- **Elemento identidad y Elemento nulo.**
 - Una identidad para un operador es un valor tal que cuando el operador se aplica a la identidad y a algún otro valor, el resultado es el otro valor.
 - 0 es la identidad para la adición: $0 + x = x + 0 = x$.
 - 1 es la identidad para la multiplicación: $1 \times x = x \times 1 = x$
 - \emptyset es la identidad para la unión: $\emptyset + L = L + \emptyset = L$
 - ϵ es la identidad para la concatenación: $\epsilon L = L \epsilon = L$
 - \emptyset es el identidad para la concatenación: $\emptyset L = L \emptyset = \emptyset$. (*Expresiones Regulares*, 2015)
- **Leyes distributivas**
 - Como la concatenación no es conmutativa, tenemos dos formas de la ley distributiva para la concatenación:
 - Ley Distributiva Izquierda para la concatenación sobre unión: $L(M + N) = LM + LN$
 - Ley Distributiva Derecha para la concatenación sobre unión: $(M + N)L = ML + NL$. (*Expresiones Regulares*, 2015)
- **Leyes de idempotencia**

- Se dice que un operador es idempotente (idempotent) si el resultado de aplicarlo a dos argumentos con el mismo valor es el mismo valor
- la suma no es un operador idempotente: $x + x \neq x$ (aunque para algunos valores si aplica como $0 + 0 = 0$)
- En general la multiplicación tampoco es idempotente: $x \times x \neq x$
- La unión de intersección son ejemplos comunes de operadores idempotentes. Ley idempotente para la unión: $L + L = L$. (*Expresiones Regulares, 2015*)

EJEMPLO

Sean r, s y t expresiones regulares. Se cumple lo siguiente:

- | | |
|--|------------------------------|
| • $r + s = s + r$. | • $(s + t)r = sr + tr$. |
| • $(r + s) + t = r + (s + t)$. | • $r + r = r$. |
| • $(rs)t = r(st)$. | • $(r^*)^* = r^*$. |
| • $\emptyset + r = r + \emptyset = r$. | • $\emptyset^* = \epsilon$. |
| • $\epsilon r = r\epsilon = r$. | • $\epsilon^* = \epsilon$. |
| • $\emptyset r = r\emptyset = \emptyset$. | • $r^+ = rr^* = r^*r$. |
| • $r(s + t) = rs + rt$. | • $r^* = r^+ + \epsilon$. |

(FaM.A.F., 2015)

BIBLIOGRAFÍA

Edgardo, Fran. (2013, July 17). 05 Lenguajes regulares. [PDF]. https://docencia.eafranco.com/materiales/teoriacomputacional/05/Clase_05.pdf

CAPTUL1.PDF. (n.d.) [Onlinedocument].
https://posgrados.inaoep.mx/archivos/PosCsComputacionales/Curso_Propedeutico/Automatas/03_Automatas_ExpresionesRegularesLenguajes/CAPTUL1.PDF

Expresiones Regulares. (2015, May 15). [PDF].
<https://ccc.inaoep.mx/ingreso/automatas/expresionesRegulares.pdf>

Juancar, Molineri. (2013, March 25). *De Autómata Finito a Expresión Regular - YouTube*. [Video file]. <https://www.youtube.com/watch>

Fa.M.A.F., Universidad Nacional de Córdoba. (2015, October 28). Expresiones Regulares. <https://wiki.cs.famaf.unc.edu.ar/lib/exe/fetch.php>. (n.d.). *Expresiones Regulares*. [PDF]. <https://wiki.cs.famaf.unc.edu.ar/lib/exe/fetch.php>

23Proceso de conversión de AFD a expresión regular | Download Scientific Diagram. (n.d.). https://www.researchgate.net/figure/Figura-123Proceso-de-conversion-de-AFD-a-expresion-regular_fig4_324114613