UNIVERSIDAD AUTÓNOMA DE CHIAPAS

FACULTAD DE CONTADURÍA Y ADMINISTRACIÓN, CAMPUS I

LICENCIATURA EN INGENIERÍA EN DESARROLLO Y TECNOLOGÍAS DE SOFTWARE

COMPILADORES

SEMESTRE: 6°   GRUPO: “M”

DR. GUTIERREZ ALFARO LUIS

ALUMNO: DANIEL ALBERTO PÉREZ ALCÁZAR

ACTIVIDAD I.- INVESTIGACIÓN Y EJEMPLOS.

Número de control: A210207

Domingo 28 de enero de 2024.

Tuxtla Gutiérrez, Chiapas.

Índice:

I.- Explicar los tipos de operadores de expresiones regulares......................... 3

II.- Explicar el proceso de conversión de DFA a expresiones regulares.......... 5

III.- Explicar leyes algebraicas de expresiones regulares. 7

Bibliografías: 9

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

Una expresión regular es una fórmula que describe un patrón de texto. Este patrón puede ser cualquier combinación de caracteres, incluyendo letras, números, símbolos, espacios en blanco y otros caracteres especiales.

Las expresiones regulares se utilizan en una variedad de aplicaciones, incluyendo:

- Compilación: Las expresiones regulares se utilizan para verificar que el código fuente de un programa cumple con las reglas de sintaxis.
- Procesamiento de lenguaje natural: Las expresiones regulares se utilizan para analizar texto y extraer información relevante
- Manipulación de texto: Se pueden utilizar para sustituir texto, editar.

Operadores Básicos:

- Ninguna o más repeticiones **(*)**: a* indica que acepta ninguna o más repeticiones de a.
- Una o más repeticiones **(+)**: a+ indica que acepta una o más repeticiones de a.
- Opcionalidad **(?)**: a? indica que a puede aparecer o no. De aparecer, sólo una vez.
- lógico **(|)**: a | b indica que existe una alternativa que puede aceptar a o b.
- Agrupación **( )**: Este operador, los paréntesis, se utiliza cuando se desea agrupar una parte de la expresión para un tratamiento especial.

Meta símbolos:

- Cualquier carácter **(.)**: .a. Indica que cualquier cadena de longitud 3 contiene la letra a en la posición del centro.
- Clase de caracteres **[ ]**: [0-9] indica cualquier carácter entre el cero y el nueve.

Comodines y abreviaturas:

\w: indica los alfanuméricos. ([A-Za-z_0-9])

\s: indica los caracteres de separación (\t ,\n, \r y espacio).

\d: indica los dígitos.

Ejemplo:

Sea ∑ un alfabeto. El conjunto ER de las expresiones regulares sobre ∑

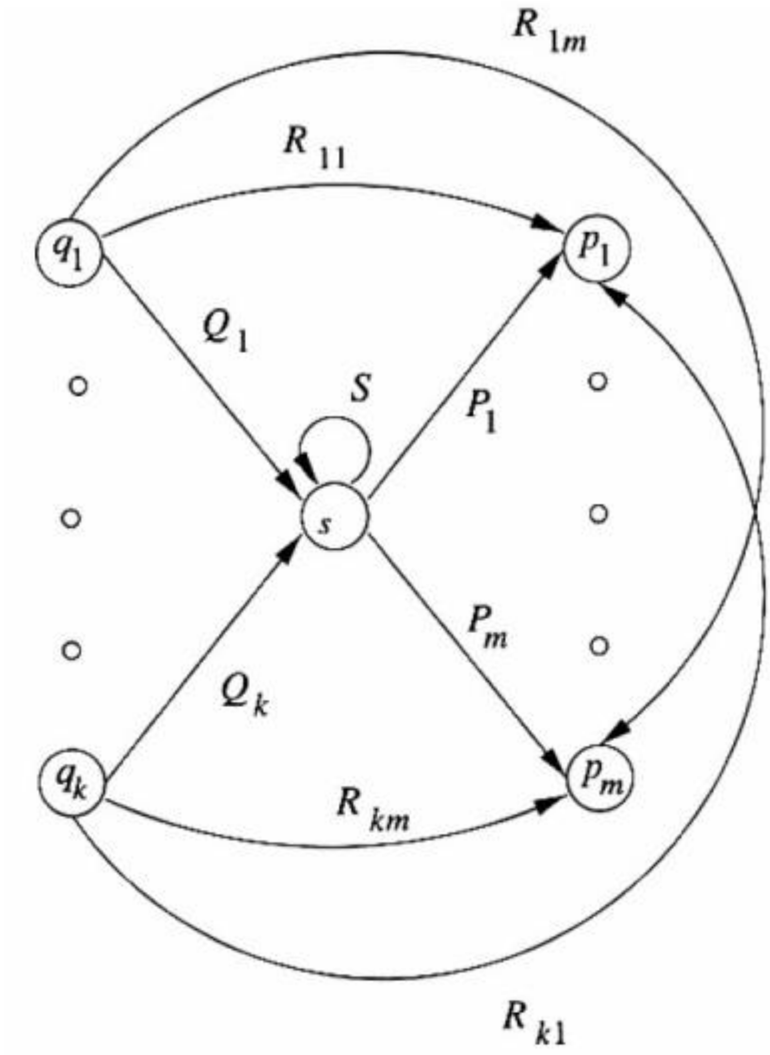
contiene las cadenas en el alfabeto ∑U {λ, |, •, *, (,), Ø} que cumplen con lo siguiente:

λ y Ø □ ER

Si ζ □ ∑, entonces ζ □ ∑ ER.

Si E1, E2 □ ER, entonces (E1 | E2) □ER, (E1 • E2) □ER, (E1)* □ER

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



Se eliminan todos los arcos que incluyen a "s"

• Se introduce, para cada predecesor qi de s y cada sucesor pj de s, una RE que representa todas las rutas que inician en qi van a s, quizás hacen un loop en s cero o más veces, y finalmente van a pj .

• La expresión para estas rutas es QiS∗Pj .

• Esta expresión se suma (con el operador unión) al arco que va de qi a pj .

• Si este arco no existe, se añade primero uno con la RE ∅

• El autómata resultante después de la eliminación de "s" es el siguiente:

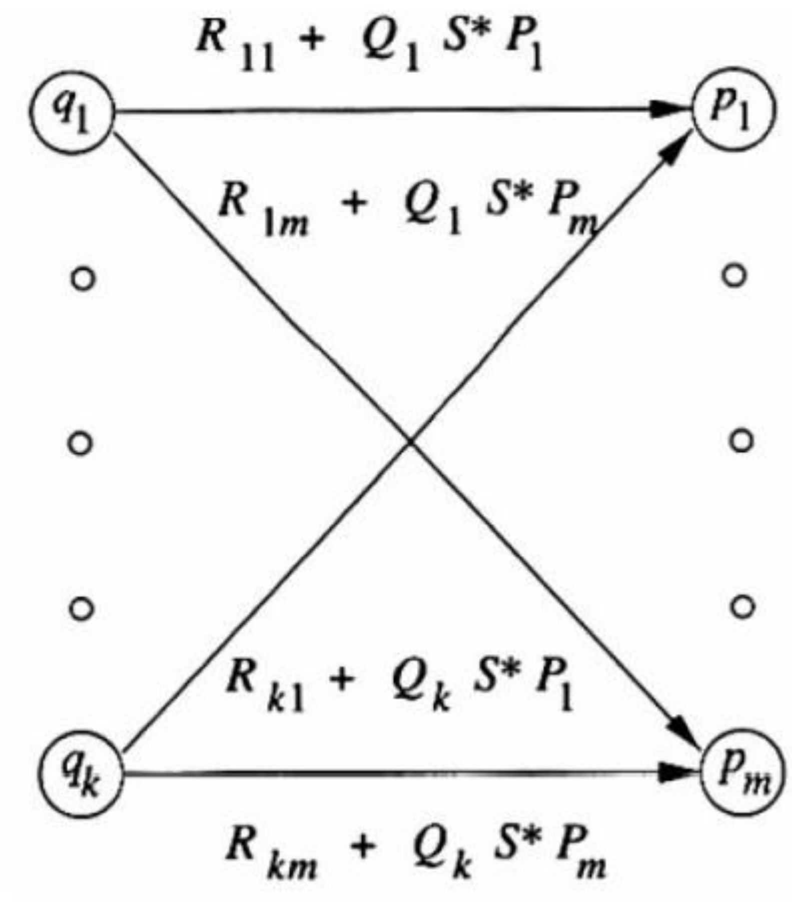


Figure 3: Eliminación de estados.

La estrategia para construir el autómata

1. Para cada estado de aceptación q, aplicar el proceso de reducción para producir un autómata equivalente con RE como etiquetas en los arcos. Eliminar todos los estados excepto q y el estado inicial q0.

2. Si q 6= q0, se genera un autómata con 2 estados como el siguiente, una forma de describir la RE de este autómata es (R + SU∗T)∗SU∗

III.- Explicar leyes algebraicas de expresiones regulares.

Las leyes algebraicas de expresiones regulares son un conjunto de reglas que se pueden utilizar para simplificar o combinar expresiones regulares. Estas leyes se basan en las propiedades de los operadores de expresiones regulares, que son la concatenación, la unión y la cerradura.

Leyes de concatenación: Las leyes de concatenación se aplican a expresiones regulares que se combinan mediante el operador de concatenación.

- **Ley asociativa:** La concatenación es asociativa, lo que significa que el orden en el que se combinan las expresiones regulares no importa. $(r1 . r2) . r3 = r1 . (r2 . r3)$
- **Ley conmutativa:** La concatenación es conmutativa, lo que significa que la secuencia de las expresiones regulares no importa.
  - $r1 . r2 = r2 . r1$
- **Ley del elemento neutro:** La concatenación del símbolo vacío con cualquier expresión regular es la propia expresión regular.
  - $\varepsilon . r = r$

Leyes de unión:

- Las leyes de unión se aplican a expresiones regulares que se combinan mediante el operador de unión.

- Ley asociativa: La unión es asociativa, lo que significa que el orden en el que se combinan las expresiones regulares no importa.
- $(r1 + r2) + r3 = r1 + (r2 + r3)$
- Ley conmutativa: La unión es conmutativa, lo que significa que la secuencia de las expresiones regulares no importa.
- $r1 + r2 = r2 + r1$
- Ley del elemento neutro: La unión del símbolo vacío con cualquier expresión regular es la expresión regular.
- $\varepsilon + r = r$
- Ley absorbente: La unión de una expresión regular con ella misma es la propia expresión regular.

$r + r = r$

Leyes de cerradura

Las leyes de cerradura se aplican a expresiones regulares que se combinan mediante el operador de cerradura.

- Ley de la cerradura: La cerradura de una expresión regular es la misma expresión regular concatenada consigo misma un número infinito de veces.
- $r^* = r \,.\, r^*$
- Ley de la cerradura vacía: La cerradura del símbolo vacío es el lenguaje vacío.

$\varepsilon^* = \varepsilon$

Bibliografías:

(N.d.). Researchgate.net. Retrieved January 29, 2024, from https://www.researchgate.net/profile/Yeiniel-Alfonso/publication/324114613_Optimizacion_de_expresiones_regulares_a_partir_del_automata_finito_equivalente/links/5abe73a245851584fa71a9c0/Optimizacion-de-expresiones-regulares-a-partir-del-automata-finito-equivalente.pdf

Regulares, 1. Expresiones. (n.d.). *Propedéutico: Teoría de Autómatas y Lenguajes Formales Expresiones regulares y lenguajes*. Inaoep.Mx. Retrieved January 29, 2024, from https://posgrados.inaoep.mx/archivos/PosCsComputacionales/Curso_Propedeutico/Automatas/03_Automatas_ExpresionesRegularesLenguajes/CAPTUL1.PDF

"Fundamentos de la teoría de autómatas" por John E. Hopcroft y Jeffrey D. Ullman (Pearson, 2007)

"Lenguajes formales y autómatas" por John E. Hopcroft, Rajeev Motwani y Jeffrey D. Ullman (Pearson, 2006)