Teoría de Lenguajes
Soluciones
1er. Parcial – Curso 2021

Ejercicio 1.- [Evaluación individual del obligatorio]

a)

```
def programa(texto):
return re.sub(r'(ja)+|muy', '', texto)
```

b)

1. r'#[\dabcdef]{6}'

2. re.findall

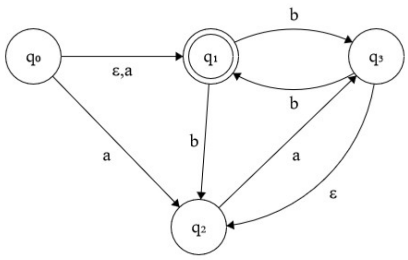
Ejercicio 2.-

Sea L_2 el lenguaje reconocido por el siguiente autómata finito $M_2=\{Q,\Sigma,\delta,q_0,F\}$ donde: $Q=\{q_0, q_1, q_2, q_3\}$ $\Sigma=\{a,b\}$ $F=\{q_1\}$ y la δ dada por:

	a	**b**	**ε**
$\mathbf{q_0}$	$\{q_1,q_2\}$	$\{\}$	$\{q_1\}$
$\mathbf{q_1}$	$\{\}$	$\{q_2,q_3\}$	$\{\}$
$\mathbf{q_2}$	$\{q_3\}$	$\{\}$	$\{\}$
$\mathbf{q_3}$	$\{\}$	$\{q_1\}$	$\{q_2\}$

a)Construya el autómata mínimo para M_2. Justifique su respuesta.
b)Dé, mediante un procedimiento formal, una expresión regular para el lenguaje L_2. Justifique su razonamiento.
c)¿Cuántas clases de equivalencia se definen según la relación R_L para el lenguaje L_2? Justifique su respuesta.

a)



Pasaje de AFND-ε a AFND

$\varepsilon\text{-cl}(q_0) = \{q_0, q_1\}$
$\varepsilon\text{-cl}(q_1) = \{q_1\}$
$\varepsilon\text{-cl}(q_2) = \{q_2\}$
$\varepsilon\text{-cl}(q_3) = \{q_3,q_2\}$

$\boldsymbol{\delta}'(q_0,a) = \varepsilon\text{-cl}(\boldsymbol{\delta}(\varepsilon\text{-cl}(q_0),a))) = \varepsilon\text{-cl}(\{q1,q2,\}) = \{q1,q2\}$
$\boldsymbol{\delta}'(q_0,b) = \varepsilon\text{-cl}(\boldsymbol{\delta}(\varepsilon\text{-cl}(q_0),b))) = \varepsilon\text{-cl}(\{q2,q3\}) = \{q2,q3\}$
$\boldsymbol{\delta}'(q_1,a) = \varepsilon\text{-cl}(\boldsymbol{\delta}(\varepsilon\text{-cl}(q_1),a))) = \varepsilon\text{-cl}(\{\}) = \{\}$
$\boldsymbol{\delta}'(q_1,b) = \varepsilon\text{-cl}(\boldsymbol{\delta}(\varepsilon\text{-cl}(q_1),b))) = \varepsilon\text{-cl}(\{q2,q3,\}) = \{q2,q3\}$
$\boldsymbol{\delta}'(q_2,a) = \varepsilon\text{-cl}(\boldsymbol{\delta}(\varepsilon\text{-cl}(q_2),a))) = \varepsilon\text{-cl}(\{q3\}) = \{q3,q2\}$
$\boldsymbol{\delta}'(q_2,b) = \varepsilon\text{-cl}(\boldsymbol{\delta}(\varepsilon\text{-cl}(q_2),b))) = \varepsilon\text{-cl}(\{\}) = \{\}$
$\boldsymbol{\delta}'(q_3,a) = \varepsilon\text{-cl}(\boldsymbol{\delta}(\varepsilon\text{-cl}(q_3),a))) = \varepsilon\text{-cl}(\{q3\}) = \{q2,q3\}$
$\boldsymbol{\delta}'(q_3,b) = \varepsilon\text{-cl}(\boldsymbol{\delta}(\varepsilon\text{-cl}(q_3),b))) = \varepsilon\text{-cl}(\{q1\}) = \{q1\}$

Como desde el estado inicial hay una transición épsilon a un estado final (q_1) entonces q_0 también es final. La tabla de transiciones del AFND queda, entonces:

	a	b
q_0	{q1,q2}	{q2,q3}
q_1	{}	{q2,q3}
q_2	{q2,q3}	{}
q_3	{q2,q3}	{q1}

F= {q0,q1}

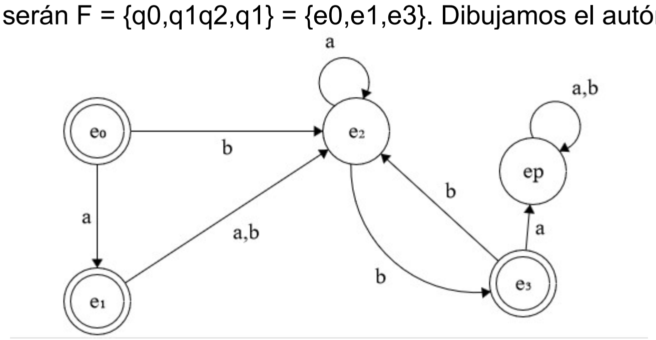
Entonces ahora hacemos el pasaje al AFD, constuyendo su nueva tabla de transición

	a	b
[q0]	[q1,q2]	[q2,q3]
[q1q2]	[q2,q3]	[q2,q3]
[q2q3]	[q2,q3]	[q1]
[q1]	-	[q2q3]

Renombramos los estados y agregamos un pozo

	a	b
e_0	e_1	e_2
e_1	e_2	e_2
e_2	e_2	e_3
e_3	e_p	e_2
e_p	e_p	e_p

Y los estados finales serán F = {q0,q1q2,q1} = {e0,e1,e3}. Dibujamos el autómata



Ahora vamos a hallar el autómata mínimo:

π_0	[e0 e1 e3] [e2 ep]
π_1	[e0] [e1 e3] [e2] [ep]
π_2	[e0] [e1] [e3] [e2] [ep]
π_3	[e0] [e1] [e3] [e2] [ep]

Se verifica que el autómata al que se llega aplicando los algoritmos para obtener el AFD ya era el mínimo.

b) Ahora mediante el método de clases de equivalencia, hallaremos una expresión regular que genere el lenguaje aceptado por el autómata. Usaremos el Lema de Arden: X = X.r | s ⇒ X = s.r*

X0 = ε
X1 = X0a
X2 = X2a | X3b | X0 b | X1 a | X1 b
X3 = X2 b
XP = XPa | XPb | X3a

X1 = X0a = ε.a = a

X2 = X2a | (X2b)b | ε.b | a.a | a.b = X2 a | X2 bb | b | aa | ab = X2(a |bb) | (b | aa | ab)
$\Rightarrow_{\text{Arden}}$ X2 = (b | aa | ab) (a |bb)*

Y ahora podemos resolver X3: X3 = X2 b = (b | aa | ab) (a |bb)* b
Hasta ahora tenemos:
X0 = ε
X1 = a
X2 = (b | aa | ab) (a |bb)*
X3 = (b | aa | ab) (a |bb)* b
nos falta hallar la expresión para el pozo

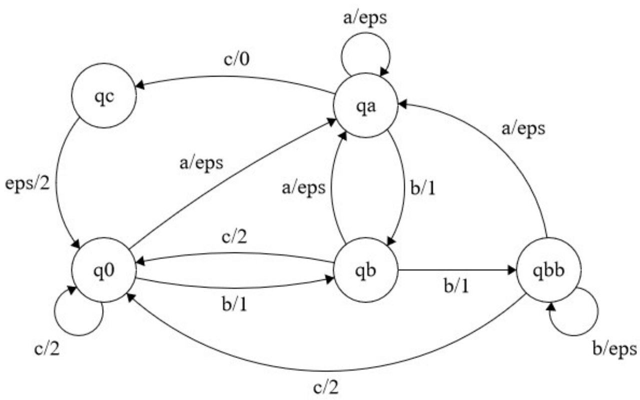
XP = XP (a|b) | (b | aa | ab) (a |bb)* ba $\Rightarrow_{\text{Arden}}$ XP = (b | aa | ab) (a |bb)* ba (a|b)*

La expresión regular será entonces la union de las expresiones regulares asociadas a estados finales, o sea

L(M) = L (X0 | X1 | X3) = L (ε | a | (b | aa | ab) (a |bb)* b)

c) Como el autómata es mínimo y completo, la cantidad de clases de RM es la misma que de RL. Por lo tanto, al tener 5 estados, se tienen 5 clases de RL.

Ejercicio 3.-



Ejercicio 4.-

a) Los siguientes lenguajes definidos sobre Σ^*, siendo $\Sigma=\{a,b,c,\#\}$ son Regulares? Justifique sus respuestas.

a.1) L_{a1} = {tiras de la forma **#x#**, con $|x|_a > |x|_b > 0$; $|x|_c \geq 1$;
$\wedge\ x \in L(c^*\,(a|b)^*)$ }

No es Regular.
Se demuestra utilizando el CR del Puping Lema para lenguajes regulares.

Sea N la cte del PL
Se elige $z = \#cb^Na^{N+1}\#$ / $z \in L_{a1}$ $|z| \geq N$ $|z|=2N+4$
Se analizan las descomposiciones de z = uvw que cumplan: $|uv| \leq N$ y $|v| \geq 1$

i) $u = \varepsilon$
$v = \#$
$w = cb^Na^{N+1}\#$

Luego, $z_i = \#^icb^Na^{N+1}\#$
Sea i = 0, luego $z_0 = cb^Na^{N+1}\#$
En este caso, falta el primer #, lo cual hace que z_0 no pertenece a L_{a1}

ii) $u = \#$
$v = cb^j$ $j \geq 0$
$w = b^{N-j}a^{N+1}\#$

Luego, $z_i = \#(cb^j)^ib^{N-j}a^{N+1}\#$
Sea i = 0, luego $z_0 = \#b^{N-j}a^{N+1}\#$
En este caso, z_0 no contiene ninguna c, lo cual hace que z_0 no pertenece a L_{a1}

iii) $u = \varepsilon$
$v = \#cb^j$ $j \geq 0$
$w = b^{N-j}a^{N+1}\#$

Luego, $z_i = (\#cb^j)^ib^{N-j}a^{N+1}\#$
Sea i = 0, luego $z_0 = b^{N-j}a^{N+1}\#$

En este caso, z_0 no sólo no comienza con un # sino que también no contiene ninguna c, lo cual hace que z_0 no pertenece a L_{a1}

iv) $u = \#cb^j$ $j+2+t \leq N$ $(j \geq 0)$
$v = b^t$ $t \geq 1$
$w = b^{N-j-t}a^{N+1}$

Luego, $z_i = \#cb^jb^{ti}b^{N-j-t}a^{N+1}\# = \#cb^{N+(i-1)t}a^{N+1}\#$
Considerando i = 2, $z_2 = \#cb^{N+t}a^{N+1}\#$
Como $t \geq 1$, la cantidad de b's es mayor o igual que la cantidad de a's. Luego z_2 no pertenece a L_{a1}.

Estas son las únicas familias que cumplen las condiciones $|uv| \leq N$ y $|v| \geq 1$, entonces, cualquier otra familia falla en alguna de estas.

Con esto se puede afirmar que L_{a1} NO es Regular

a.2) $L_{a2} = \{ a^pb^ra^k$; con $p>0, r\geq 0, k>0$ k par$\}$

Es Regular. En base a las restricciones de las cantidades de los símbolos y teniendo en cuenta que no hay relaciones entre ellos:
- al menos una a
- luego pueden o no venir b's
- finalmente una cantidad par de a's, pero al menos dos.

Se puede construir entonces la ER **$aa^*b^*aa(aa)^*$**

b) Sea el lenguaje $L_b = \{ cb^ka^pc$; con $k\geq 0, p>0 \}$ y sea el siguiente homomorfirmo
h: $\{a,b,c\}^* \rightarrow \{0\}^*$ / $h(a) = 0$, $h(b) = 0$, $h(c) = 0$
Exprese $h(L_b)$ con la menor cantidad de operadores de expresiones regulares. Explique su razonamiento.

Nuevamente, como se trata de un lenguaje donde las restricciones de los símbolos del lenguaje no guardan relación, puede observarse que el lenguaje puede expresarse mediante la siguiente ER: **cb^*aa^*c**

Aplicando la definición de homomorfirsmo y su extensión para lenguajes se tiene:

$h(cb^*aa^*c) = h(c)h(b^*)h(a)h(a^*)h(c) = 00*00*0 =$ **0000*** es la ER con menor cantidad de operadores, 3 concatenaciones y una clausura de kleene.

c) ¿La siguiente afirmación es Verdadera o Falsa? Justifique su respuesta.
Si L_d no es un lenguaje regular, existe un lenguaje regular no vacío L_c, tal que $L_c \subseteq L_d$.

Verdadera. Considere cualquier tira $w \in L_d$ (que se sabe que no es vacío porque no es regular) y defina $L_c = \{w\}$.
Como L_c está compuesto por un único elemento, es finito y por lo tanto regular y además, $L_c \subseteq L_d$