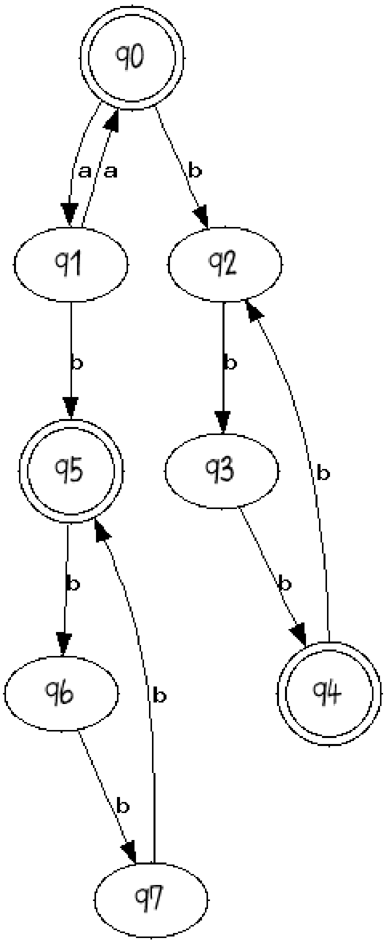
Teoría de Lenguajes
Teoría de la Programación I
Soluciones

Ejercicio 1

a) Construya un autómata mínimo M_1 / $L_1 = L(M_1)$

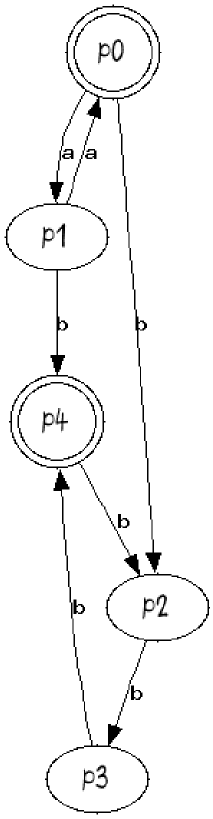
Autómata para el lenguaje:



Particiones para minimizar:

p0:[q0,q4,q5] p1:[q1,q2,q3,q6,q7]
p0:[q0,q4,q5] p1:[q1,q3,q7] p2:[q2,q6]
p0:[q0] p4:[q4,q5] p1:[q1] p3:[q3,q7] p2:[q2,q6]

Autómata mínimo:



b) Construya una gramática simplificada G_1 / $L_1 = L(G_1)$. Justifique.

S -> ε | aP_1 | bP_2
P_1 -> aS | bP_4
P_4 -> bP_2 | ε
P_2 -> bP_3
P_3 -> bP_4

Todas son positivas y alcanzables, ninguna producción unitaria.

Elimino producciones epsilon:

S -> ε | aP_1 | bP_2
P_1 -> aS | bP_4 | b
P_4 -> bP_2
P_2 -> bP_3
P_3 -> bP_4 | b

c) ¿Cuántas clases de equivalencia tiene L_1 según R_L? Justifique.

Por teorema de MyHill-Nerode, las clases de equivaliencia según RL coinciden con las clases de equivalencia del autómata mínimo (una clase por cada estado). Dado que el autómata mínimo posee 5 estados + 1 estado pozo (no representado en el diagrama), existen 6 clases de equivalencia.

d) Dé una expresión regular r_1 / $L_1 = L(r_1)$. Justifique.
dado a cada una de estas clases.

Planteo sistema de ecuaciones:

$X_0 = \varepsilon \mid X_1a$
$X_1 = X_0a$
$X_2 = X_4b \mid X_0b$
$X_3 = X_2b$
$X_4 = X_3b \mid X_1b$

Resuelvo:

$X_0 = \varepsilon \mid X_0aa$
$X_0 = (aa)^*$

$X_4 = X_2bb \mid X_0ab$
$X_4 = X_4bbb \mid X_0bbb \mid X_0ab$
$X_4 = X_4bbb \mid X_0 (bbb \mid ab)$
$X_4 = (aa)^*(bbb \mid ab)(bbb)^*$

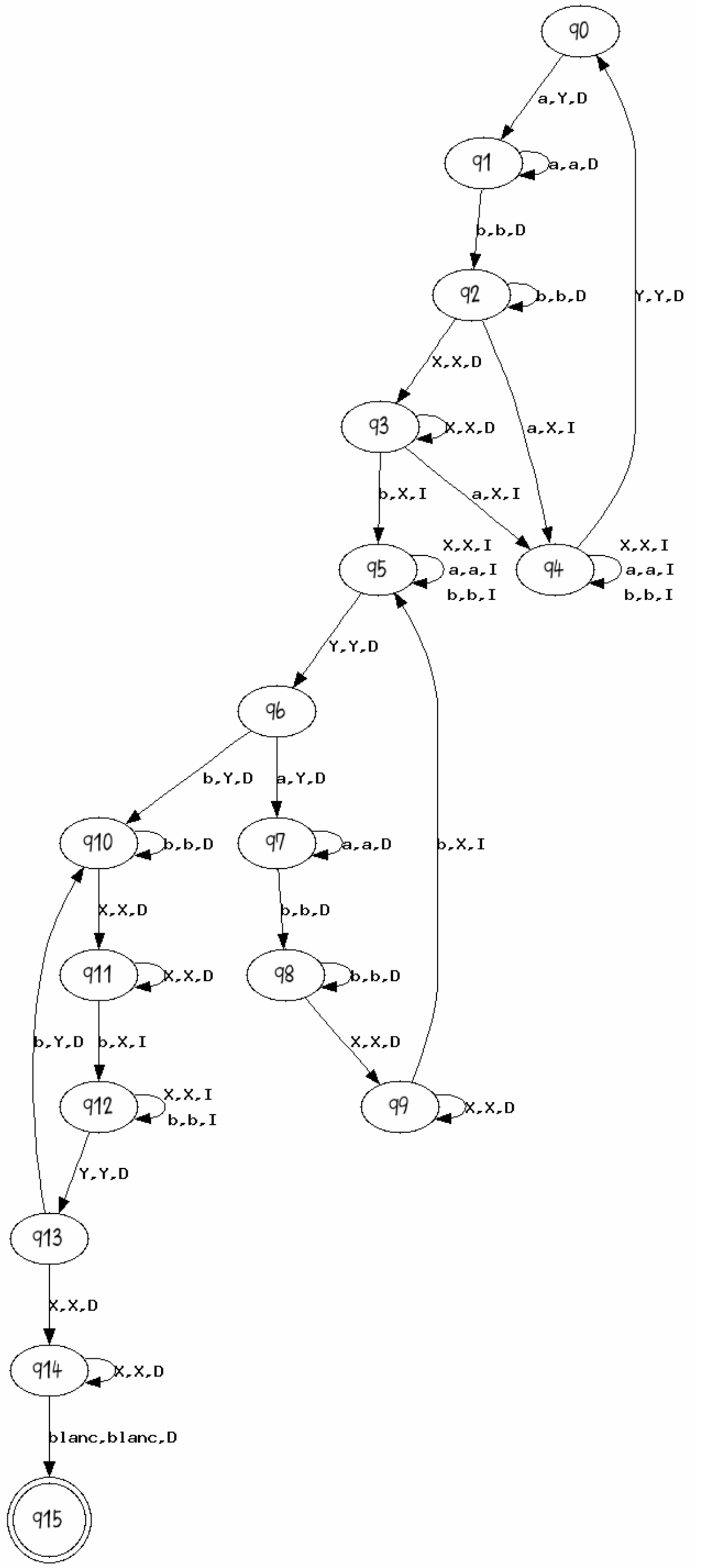
Expresión regular del lenguaje = unión de expresiones de para estados finales:

$X_0 \mid X_4 = (aa)^* \mid (aa)^*(bbb \mid ab)(bbb)^*$

Ejercicio 2

Sea $L_2 = \{ a^p a^q b^r a^p b^q b^r \ / \ p,q,r>0\}$

a) Construya un autómata M_2 / $L_2 = L(M_2)$



Nota: Las gramáticas y los autómatas deben corresponderse con el tipo del lenguaje considerado en cada caso, según la Jerarquía de Chomsky. Se valora positivamente la simplicidad de las soluciones propuestas así como una breve explicación de éstas. Todas las respuestas deben estar debidamente justificadas.

b) Clasifique a L_2 según la Jerarquía de Chomsky. Justifique

El lenguaje es recursivamente enumerable porque en la parte a) se construyó una Máquina de Turing que lo reconoce. Demostraremos que no es libre de contexto, utilizando el CR del PL. Para eso elegimos la tira $z=a^{2N}b^Na^Nb^{2N}$, siendo N la constante del Pumping Lemma.

Consideramos todas las descomposiciones posibles de la tira z en uvwxy que cumplen $|vx|>0$ y $|vwx|\leq N$, y probamos que existe un i tal que uv^iwx^iy no pertenece a L_2

Familias posibles

```
     aaaaaaa ...   aaaaabbbb ...  bbbbaaaa ... aaaaabbbb .... bbbb
1        vx
2                     vxxxx
3                     vvvxxx
4                              vx
5                                    vxxxx
6                                    vvvvxx
7                                          vx
8                                                vxxxx
9                                                vvvvxx
10                                                        vx
11          v              x
12                            v                   x
13                                       v               x
```

Familia 1:
$u=a^j$
$v=a^k$
$w= a^l$
$x= a^m$
$y= a^{2N-j-k-l-m}b^Na^Nb^{2N}$

con $k+m>0$

$z_0= a^{2N-k-m}b^Na^Nb^{2N}$, que no pertenece al lenguaje pues al ser $k+m>0$, la suma de las primeras a's y b's es diferente a la suma de las segundas.

Familia 2:
$u=a^j$
$v=a^k$
$w= a^l$
$x= a^{2N-j-k-l}b^m$
$y= b^{N-m}a^Nb^{2N}$

Eligiendo i=2, $z_i= a^j\ a^{2k}a^l(a^{2N-j-k-l}b^m)^2b^{N-m}a^Nb^{2N}= a^{2N+k}b^ma^{2N-j-k-l}b^Na^Nb^{2N}$, que no pertenece al lenguaje porque se pierde el ordenamiento de secuencias de a's y b's necesario para que las tiras pertenezcan.

Las familias 3,5,6,8,9 son análogas: eligiendo i=2 z_i no pertenece al lenguaje porque se pierde el ordenamiento de secuencias de a's y b's necesario para que las tiras pertenezcan.

En las familias 4,7,10,11,13: análogo a la familia 1, eligiendo i=0 la suma de las primeras a's y b's es diferente a la suma de las segundas.

En la familia 12:

$u=a^{2N}b^{j}$
$v=b^{k}$
$w= b^{N-j-k}a^{m}$
$x= a^{n}$
$y= a^{N-n-m}b^{2N}$

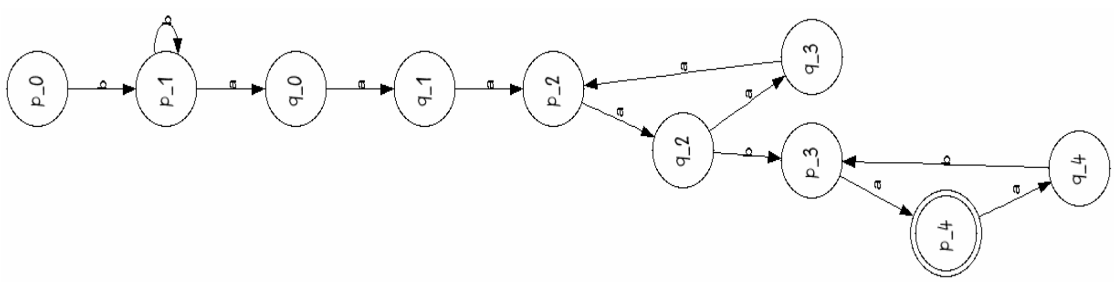
Con $k+n>0$

Si $k\neq n$, la suma de las primeras a's y b's es diferente a la suma de las segundas, cuando $i=0$.
Si $k=n>1$, elijo $i=N+1$, y $z_N=a^{2N}b^{j}b^{k(N+1)}b^{N-j-k}a^{m}a^{k(N+1)}a^{N-m-k}b^{2N}= a^{2N}b^{N(k+1)}a^{N(k+1)}b^{2N}$, pero $N(k+1) \geq 2N$ (porque $k>0$), y por lo tanto la tira no pertenece al lenguaje porque la segunda secuencia de b's debe ser estrictamente mayor que la primera.

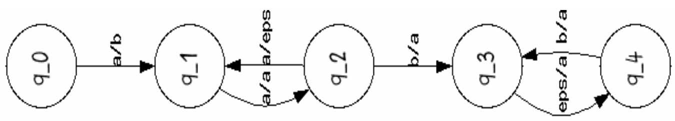
Como estas son todas las descomposiciones posibles de z en uvwxy que cumplen $|vx|>0$ y $|vwx|\leq N$, por el CR del PL el lenguaje **no** es recursivamente enumerable.

Ejercicio 3

a) Construya un autómata de dos cintas que acepte al lenguaje $\{(a^{2k}b^{p}, b^{t}a^{2p+k}) / k, p, t>0\}$



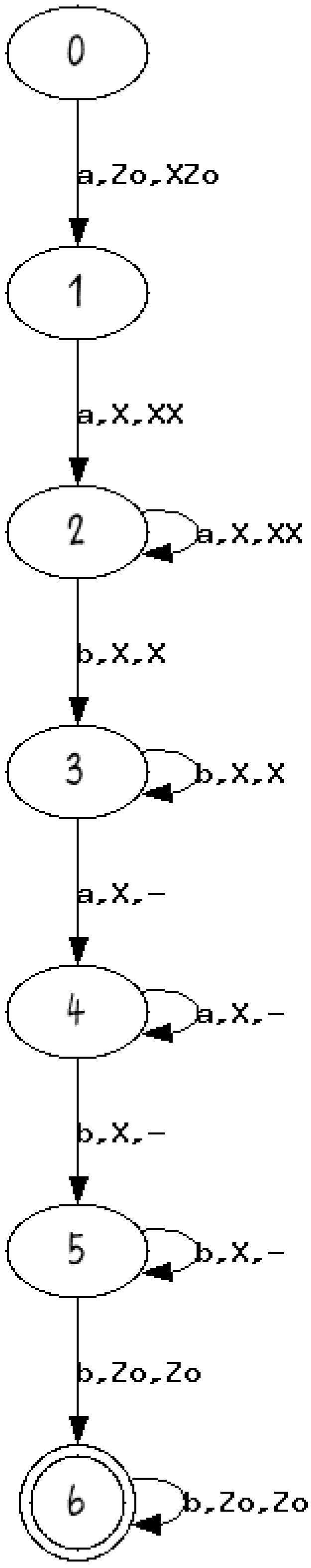
b) Construya una máquina de Mealy tal que, para la entrada $a^{2k}b^{p}$, devuelva la salida ba^{2p+k} con $k, p> 0$



Ejercicio 4 [Teoría de Lenguajes]

Sea $L_3 = \{ a^p a^q b^r a^p b^q b^s \,/\, p,q,r,s>0\}$

a) Construya un autómata M_3/ L_3= L(M_3)



b) Construya una gramática G_3/ L_3= L(G_3)

S $\rightarrow$ Sb | S'b
S' $\rightarrow$ aS'b | aS''b
S'' $\rightarrow$ aS''a | aS''a
S''' $\rightarrow$ S'''b | b

c) Clasifique a L_3 según la Jerarquía de Chomsky. Justifique

Es Libre de contexto por que pudimos construir un APD y una GLC en las partes anteriores que lo reconocen.
Y no es regular. Esto lo demostraremos aplicando el Contra-reciproco del PL para lengujes regulares.

Dado n constante cualquiera, elijo z = $a^n aba^n bb \in L_3$ y $|z|=2n+4 \geq n$

Estudiamos todas las descomposiciones de z = uvw talque $|uv| \leq n$ y $|v|\geq 1$
y trataremos de encontrar un $i\geq 0$ para c/descomposición tal que $uv^iw \notin L_3$

Familia	a^{n+1}	b	a^n	b^2	
Nro. 1	v				

La única familia de descomposiciones que cumple $|uv|\leq n$ y $|v|\geq 1$ es la familia 1, donde:
$u = a^p$
$v = a^q$ con $p+q\leq n$ y $q\geq 1$
$w = a^{n+1-p-q}ba^nb^2$
$uv^iw = a^p(a^q)^i a^{n+1-p-q}ba^nb^2 = a^{n+1+q(i-1)}ba^nb^2$
Eligiendo i=2 vemos que $uv^iw \notin L_3$, porque la cantidad de as y bs de la primer parte: n+1+q+1; no se corresponde con cantidad de as y bs de la segunda parte de la tira: n+2.

Al haber estudiado todas las descomposiciones posibles para la tira z, hemos demostrado que L_3 no puede ser regular.

d) ¿Cuántas clases de equivalencia tiene L_3 según R_L? Justifique.

Infinitas.
Por el teorema de "*Myhill*-Nerode" sabemos que la cantidad de clases de equivalencias R_{L3} es finita si y solo si el lenguaje es regular, pero como L_3 no es regular, entonces la cantidad de clases de equivalencias de R_{L3} es infinita.

Ejercicio 4 [Teoría de la Programación 1]

i. **Falso**. Consideremos al conjunto de los naturales N.

$$f_N(n, m) = \begin{cases} 1 & \text{si } (n \in N \wedge \langle I_x(n), m\rangle\downarrow) \equiv \text{ si } \langle I_x(n), m\rangle\downarrow \\ 0 & \text{en caso contrario} \end{cases}$$

Luego f_N es la función STOP que, como se probó en el teórico, no es una función computable.

ii. **Verdadero**. Como A es un conjunto r.e., su función característica parcial es computable; sea MA la macro que la computa.

Construimos al siguiente programa en P de índice p:

```
PROGRAM(X0)
   X1:= FST(X0);              -- extraigo n
   X2:= SND(X0);              -- extraigo m
   X3:= MA(X1);               -- ¿n ∈ A?
   X4:= EVAL_PROG(X1,X2)      -- no entró en loop en el paso previo, pruebo <Ix(n), m>↓
   X5:= SUC(0);               --  n ∈ A y  eval_prog terminó => retorno 1
RESULT(X5)
```

Se cumple que:

$$I_x(p,\langle n,m\rangle)\downarrow \Leftrightarrow^1 MA(n)\downarrow \wedge \text{EVAL\_PROG}(n,m)\downarrow \Leftrightarrow^2 n \in A \wedge \langle Ix(n), m\rangle\downarrow \Leftrightarrow^3 g_A(n, m) = 1$$

1. Por construcción de p.
2. Por definición de MA y EVAL_PROG.
3. Por definición de g_A

Además, por construcción se cumple que: $I_x(p,\langle n,m\rangle)\downarrow \Leftrightarrow I_x(p,\langle n,m\rangle)\Rightarrow 1$

En definitiva:

$$I_x(p, \langle n,m\rangle)\Rightarrow 1 \Leftrightarrow n \in A \wedge \langle Ix(n), m\rangle\downarrow \Leftrightarrow gA(n, m) = 1,$$
$$I_x(p, \langle n,m\rangle)\uparrow \Leftrightarrow n \notin A \vee \langle Ix(n), m\rangle\uparrow \Leftrightarrow gA(n, m) = \text{indef.}$$

Luego, el programa de índice p computa a la función gA, qed.

iii. **Verdadero**. Consideremos al conjunto vacío.

$$f_\varnothing(n, m) = \begin{cases} 1 & \text{si } n \in \varnothing \wedge \langle Ix(n), m\rangle\downarrow \equiv \text{ falso} \\ 0 & \text{en caso contrario} \end{cases}$$

Luego $f_\varnothing$ es la función constante cero, computada por el siguiente programa en P:

```
PROGRAM(X0)
      X1:=0
RESULT(X1)
```

Ejercicio 5 [Teoría de la Programación 1]
Ver Garey & Johnson.