(1) Defina gramáticas regulares que generen los siguientes lenguajes:

(a) Números enteros. (ej. 20, -344, -03).

(b) Números enteros pares sin ceros no significativos (ej. **no** puede generar -02).

(c) Expresiones decimales de números racionales (ej. -3.1415926, 0.0001)

(d) $\{\alpha \in \Sigma^* \mid \alpha = a^n b c^m,\ n, m > 0\},\ \Sigma = \{a, b, c\}$.

a-

$S \longrightarrow 1N \mid 2N \mid \ldots \mid 0N \mid -N$

$T \longrightarrow 1T \mid 2T \mid \ldots \mid 0T \mid \varepsilon$ $\longrightarrow$ N con ε

$N \longrightarrow 1T \mid 2T \mid \ldots \mid 0T$ $\longrightarrow$ N sin ε

b-

$S \longrightarrow 2N \mid 4N \mid 6N \mid \ldots \mid -N \mid 0C$ $\longrightarrow$ Propongo agregar la variable C para la opcion del cero, ya que es par. Y para forzar a no poder colocar ceros no significativo

$T \longrightarrow 2T \mid 4T \mid 6T \mid 0T \mid \varepsilon$

$N \longrightarrow 2T \mid 4T \mid 6T$

$C \longrightarrow \varepsilon$

c-

$S \longrightarrow 1N \mid 2N \mid 3N \mid 0N \mid -N$

Ingreso una variable nueva puesto que quiero forzar, a que si coloco un punto, no volver a tener la posibildad de colocarlo de nuevo, ya que seria incorrecto como lenguaje.

$T \longrightarrow 1T \mid 2T \mid 3T \mid 0T \mid \varepsilon$

$N \longrightarrow 1T \mid 2T \mid 3T \mid 0 \mid .P$

$P \longrightarrow 0P \mid 1P \mid 2P \mid \ldots \mid \varepsilon$

d-

(2) Sea G la gramática con símbolo inicial S y derivaciones

$$S \to bS \mid aA \mid a, \qquad A \to aS \mid bB, \qquad B \to bA \mid aS \mid b$$

*No es regular

(donde a y b son los símbolos terminales).

(a) Demuestre, proporcionando la derivación correspondiente, que las siguientes cadenas pertenecen a $L(G)$

$$aaabb, \qquad bbbaaaa, \qquad abaaabbabbaa.$$

(b) Probar que en el lenguaje $L(G)$ todas las cadenas tienen un número impar de símbolos a.

(c) ¿Es $L(G)$ el lenguaje formado por todas las cadenas con un número impar de símbolos a?

- Si $X \xrightarrow{*} \beta$ y $\beta \in (V \cup T)^* \setminus T^*$ entonces $\beta = \alpha Y$ con $\alpha \in T^*$ y $Y \in V$. Más aún, si $\alpha = \epsilon$ entonces $Y = X$.
- Si $X \xrightarrow{*} \alpha$ y $\alpha \in T^*$ entonces existe $Y \in V$ tal que $X \xrightarrow{*} \alpha Y$ y $Y \longrightarrow \epsilon$ está en P (y luego $X \xrightarrow{*} \alpha Y \Rightarrow \alpha$).

a- $S \longrightarrow aA \longrightarrow aaS \longrightarrow aaaS \longrightarrow aaabB \longrightarrow aaabb$ ✓

$S \longrightarrow bS \longrightarrow bbS \longrightarrow bbbS \longrightarrow bbbaA \longrightarrow bbbaaS \longrightarrow bbbaaaA \longrightarrow bbbaaaaS \longrightarrow bbbaaaa$ ✓

(la ultima a cargo del lector)

b- Probemos por induccion en k, que se cumple lo siguiente:

i) $S \Rightarrow^k \alpha \in T^*$, entonces $\alpha \in Li$

ii) $A \Rightarrow^k \alpha \in T^*$, entonces $\alpha \in Lp$

iii) $B \Rightarrow^k \alpha \in T^*$, entonces $\alpha \in Lp$

i) S --> |a| = impar
ii) A --> |a| = par
iii) B --> |a| = impar

* Definamos $Li = \{\alpha \in T^* / |\alpha|_a \text{ es impar}\}$

$Lp = \{\alpha \in T^* / |\alpha|_a \text{ es par}\}$

*Caso base: $\boxed{k=0}$

$\longrightarrow (S - 0k \in T) \longrightarrow 0k \in Li \equiv$ Falso $\longrightarrow 0k \in Li \equiv$ Verdadero

$\longrightarrow$ Trivial con los demás

*Caso inductivo: $\boxed{k = k+1}$

$\longrightarrow S \longrightarrow^{(k+1)} \alpha \equiv S \xrightarrow{k+1} \alpha \equiv S \longrightarrow \gamma \xrightarrow{k} \alpha$ ▲

▲ $S \to \gamma$, luego hay α', α'', $W \to \rho$ tq. $S = \alpha' W \alpha''$, $\gamma = \alpha' \gamma \alpha''$

luego, $|W| = 1$ puesto que $\in V$, entonces $\alpha' = \alpha'' = \varepsilon$, $W = S$ y $\gamma = \rho$. Luego γ debe de ser, bS - aA o a.

* $bS \xrightarrow{k} \alpha$, $\alpha = b\alpha$ y $S \xrightarrow{k} \alpha'''$. Por i) ✓

* $aA \xrightarrow{k} \alpha$, $\alpha = a\alpha$ y $A \xrightarrow{k} \alpha'''$. Por ii) ✓

* $a \xrightarrow{k} \alpha$, $\alpha = a$ y $k = 0$ puesto que a no tiene variable, luego $\alpha = a \in Li$.

(3) Sea G la gramática regular definida por las producciones

$$S \to bS \mid aA \mid bE, \qquad A \to aS \mid bA \mid aE, \qquad E \to \epsilon$$

(donde a y b son los símbolos terminales). Demuestre que $\alpha \in L(G)$ sii $\alpha \neq \epsilon$ y contiene un número par de símbolos a.

* $S \longrightarrow \alpha$, $|\alpha|_a = \text{par}$
* $A \longrightarrow \alpha$, $|\alpha|_a = \text{impar}$

Nuevamente, hacemos inducción en $|\alpha|$.

→ $|\alpha| = 1$, luego $S \longrightarrow bE \longrightarrow b$ y $A \longrightarrow aE \longrightarrow a$ ✓

→ $\alpha = a\beta$. Caso aA ⇒ $S \longrightarrow aA \longrightarrow a\beta$, luego 1 + impar a's = par ✓

"a" ; $\underbrace{A \longrightarrow a\beta}_{\text{impar}}$ → $|a\beta|_a = \text{par}$.

. Caso bS ⇒ $S \longrightarrow bS \longrightarrow aA \longrightarrow ba\beta$ → par nuevamente

. Caso bE ⇒ $S \longrightarrow bE \longrightarrow b$ → par

(4) Sea L_1 (respectivamente L_2) el lenguaje generado por la gramática del Ejercicio 2 (respectivamente, Ejercicio 3). Encuentre gramáticas regulares que generen los lenguajes L_1 y L_1L_2 (una gramática para cada lenguaje).

Es facil hacer que la gramatica del ejercicio 2 sea regular, agregando una produccion mas, tal que:

* $S \longrightarrow bS \mid aA \mid aE$, $A \longrightarrow aS \mid bB$, $B \longrightarrow bA \mid aS \mid bE$, $E \longrightarrow \varepsilon$

* Para L_1L_2, primero en L_1: $S_1 \longrightarrow bS_1 \mid aA_1 \mid aS_2$, $A_1 \longrightarrow aS_1 \mid bB_1$, $B_1 \longrightarrow bA_1 \mid aS \mid bS_2$ y en L_2:

$S_2 \longrightarrow bS_2 \mid aA_2 \mid bE$, $A_2 \longrightarrow aS_2 \mid bA_2 \mid aE$, $E \longrightarrow \varepsilon$

(5) Dada la expresión regular

$$b(a^* + b)^* bb^* a$$

construir una gramática que genera exactamente el lenguaje que denota la expresión regular. ¿Es regular? Si no le es, cómo la modificaría para que lo sea?

Observemos que la expresion regular denota un lenguaje de cadenas que empiezan solamente en "b", que luego tienen un epsilon, un bucle en a, o una b, seguida de una fija, y una cierta cantidad de b's y termina con una "a" fija.

→ b... " " b " " a

* $S \longrightarrow bEbBa$
  $E \longrightarrow aE \mid bE \mid \varepsilon$
  $B \longrightarrow bB \mid \varepsilon$

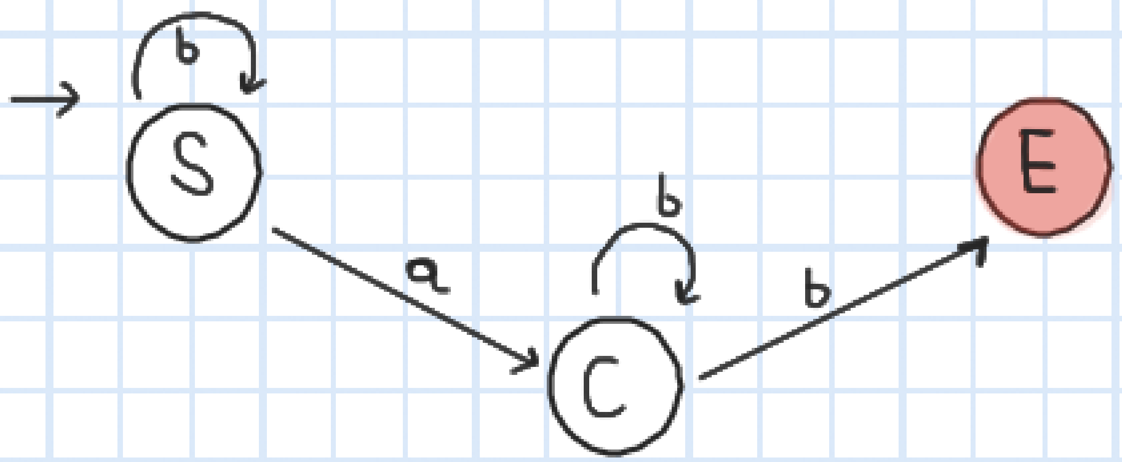
"Gramatica libre de contexto" (no regular)

→ Ahora, la regular:

$S \longrightarrow bE \mid$
$E \longrightarrow aE \mid bE \mid bT$
$T \longrightarrow bT \mid aR$
$R \longrightarrow \varepsilon$

(6) Obtener un autómata finito no determinístico que acepte únicamente las cadenas generadas por la siguiente gramática regular G: las variables son S, E y C, con S como variable inicial; las constantes son a, b y las producciones son

$$S \to bS, \quad S \to aC, \quad C \to bC, \quad C \to bE, \quad E \to \epsilon.$$



(7) Sea G la gramática definida por las producciones

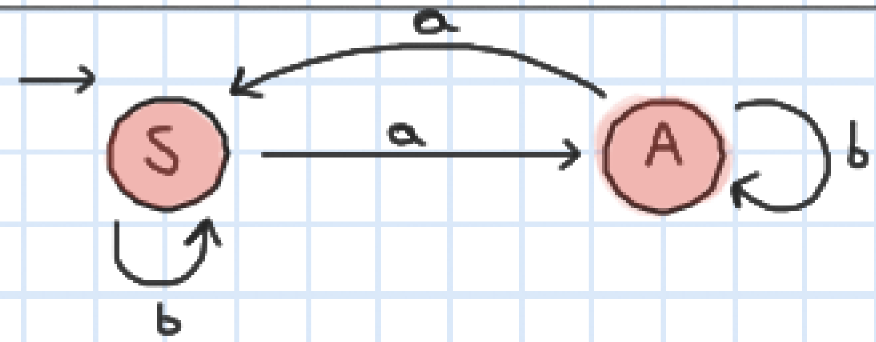
$$S \to bS \mid aA \mid b, \qquad A \to aS \mid bA \mid a$$

(donde a y b son los símbolos terminales). Obtener:

(a) Un autómata finito no determinístico M tal que $L(M) = L(G)$.

(b) Una expresión regular R tal que $L(R) = L(G)$.

a_



* G no es regular, entonces la hagamos Regular:

$S \to bS \mid aA \mid bE$, $A \to aS \mid bA \mid bE$

$E \to \epsilon$

b_ Usando Kleene: $L(M) = L_{SS}(Q) + L_{SA}(Q)$

* $L_{SS}(Q) = I_S(Q)^* = a L_{AA}\{A\} a + b^*, (ab^*a + b^*)^*$

* $L_{SA}(Q) = I_S(Q)^* F_{SA}(Q) = (ab^*a + b^*)^* F_{SA}(Q)$, $F_{SA}(Q) = a L_{AA} = a I_A^*(Q) = ab^*, (ab^*a + b^*)^* ab^*$

$L(M) = (ab^*a + b^*)^* + (ab^*a + b^*)ab^*$