EJERCICIO 1: INTRODUCCIÓN Y GRAMÁTICAS.

EXPLICAR: ¿en qué consiste la Jerarquía de Chomsky?

Clasificación de gramáticas (Jerarquía de Chomsky)

Las gramáticas pueden clasificarse según el formato de sus producciones.

Si un lenguaje es generado por una gramática de tipo X, ,entonces el lenguaje es de tipo X.

- **Gramáticas tipo 0**

  Son las más generales. (**Gramáticas sin restricciones**).Las producciones pueden ser de cualquier tipo permitido, es decir de la forma $\alpha \to \beta$ con $\alpha \in (N \cup \Sigma)^{+}$ y $\beta \in (N \cup \Sigma)^{*}$

  Los lenguajes generados por estas gramáticas son los ***Lenguajes con Estructura de Fras**e* (se agrupan en la clase L_0) Estos lenguajes también se conocen como ***lenguajes recursivamente enumerables***.

- **Gramáticas tipo 1**

  Gramáticas sensibles al contexto.

  Las producciones son de la forma $\alpha \to \beta$ con $|\alpha| \leq |\beta|$, excepto para $S \to \lambda$.

  Los lenguajes generados por estas gramáticas son los **lenguajes sensibles al contexto.** (L_1)

- **Gramáticas tipo 2**

  Gramáticas libres del contexto.

  Las producciones son de la forma $A \to \alpha$.

  Los lenguajes generados por estas gramáticas son los **lenguajes libres de contexto** (L_2).

- **Gramáticas tipo 3**

  Gramáticas regulares.

Lineales por derecha:	*Lineales por izquierda:*
$\boxed{\begin{array}{l} A \to bC \\ A \to b \\ A \to \lambda \end{array}}$	$\boxed{\begin{array}{l} A \to Cb \\ A \to b \\ A \to \lambda \end{array}}$

EJERCICIO 2: AUTÓMATAS FINITOS.

DEMOSTRAR: "Si L es aceptado por algún AFD entonces L es aceptado por algún AFND-λ".

1ª Demostración. Si L es aceptado por algún AFD, entonces es aceptado por un AFN-λ.

Transformamos D en un AFND-λ añadiendo transiciones $\forall q \in Q_D : \delta_E(q,\lambda) = \varnothing$ y $\forall q \in Q_D \forall a \in \Sigma : \delta_D(q,a) = p \Rightarrow \delta_E(q,a) = \{p\}$

Queda: $E = (Q_E, \Sigma, \delta_E, \{q_D\}, F_E)$ con $Q_E = \{\{q\} / q \in Q_D\} \cup \varnothing$ y $F_E = \{\{q\} / q \in F_D\}$

Como las transiciones no han cambiado, toda palabra aceptada en el AFD es aceptada en el AFND-λ

EJERCICIO 3: AUTÓMATAS FINITOS/ GRAMÁTICAS LIBRES DE CONTEXTO/ LLC/ANÁLISIS SINTÁCTICO.

SELECCIONA EN CADA CASO LA OPCIÓN CORRECTA:

I. EL LENGUAJE ACEPTADO POR UN AUTÓMATA FINITO NO DETERMINISTA ES:

A. $L = \{\omega \in \Sigma^* | \hat{\delta}(q_0, \omega) \in F\}$

B. $L = \{\omega \in \Sigma * | \hat{\delta}(q_0, \omega) \cap F \neq \emptyset\}$

C. $L = \{\omega \in \Sigma * | \hat{\delta}(q_0, \omega) \cap \bar{F} = \emptyset\}$

D. $L = \{\omega \in \Sigma * | \hat{\delta}(q_0, \omega) \subset F\}$

II. UNA GRAMÁTICA ES AMBIGUA SI:

A. para una cadena α existen dos árboles distintos con S como raíz y α en la base.

B. para una cadena α existen dos derivaciones más a la izquierda distintas desde S.

C. A y B son correctas.

D. Ninguna opción es correcta.

III. EL LEMA DE BOMBEO PARA LENGUAJES LIBRES DE CONTEXTO SE DEMUESTRA USANDO:

A. Una gramática en Forma Normal de Chomsky.

B. Una gramática en Forma Normal de Greibach.

C. Una gramática sin recursividad a izquierda.

D. Una gramática factorizada por izquierda.

IV. EN EL ANÁLISIS SINTÁCTICO ASCENDENTE, CUANDO ACCION[estado, a] ES "desplazar t":

A. Se sacan $|\beta|$ símbolos de la pila.

B. Se mete el símbolo t de la entrada en la pila.

C. Se sacan $|\beta|$ símbolos de la pila y se consume 'a'.

D. Se mete el símbolo t de la entrada en la pila y se consume 'a'.

EJERCICIO 4: ***EXPRESIONES REGULARES/ LENGUAJES REGULARES/ ANÁLISIS SINTÁCTICO.***

INDICA SI LAS SIGUIENTES AFIRMACIONES SON VERDADERAS O FALSAS. JUSTIFICAR TODO BREVEMENTE.

I. La clausura * de $\varnothing$ es $\varnothing$. V
II. La INTERSECCIÓN de dos lenguajes regulares no es regular. F
III. Existen métodos de análisis sintáctico descendente que no requieren backtracking. √

1) $\alpha^* = \alpha^+ + \lambda$

2) **Intersección.**

Dados dos lenguajes regulares, su **INTERSECCIÓN** también es un lenguaje regular.

Si L y M son lenguajes regulares, entonces también lo es $L \cap M$.

Dado que ya se demostró que tanto la unión como el complemento de lenguajes regulares es regular, la intersección debe ser regular por cumplirse $L \cap M = \overline{\overline{L} \cup \overline{M}}$.
Pero también existe una forma directa de obtener la intersección:

Demostración

Sea L un lenguaje regular, reconocido por el AFD $A_L =< Q_L, \Sigma, \delta_L, F_L >$ y M un lenguaje regular, reconocido por el AFD $A_M =< Q_M, \Sigma, \delta_M, F_M >$

Se define el autómata $A =< Q_L \times Q_M, \Sigma, \delta, (q_L, q_M), F_L \times F_M >$ (Donde $\times$ es el producto cartesiano) con la función de transición definida de la siguiente manera:

$$\delta\big((p,q),a\big) = (\delta_L(p,a), \delta_M(q,a))$$

Para ver que $L(A) = L(A_L) \cap L(A_M)$, $\forall\, \omega \in \Sigma^*: \omega \in L(A) \Leftrightarrow \hat{\delta}\big((q_L, q_M), \omega\big) \in F_L \times F_M$

Pero eso equivale a decir que $\hat{\delta}\big((q_L, q_M), \omega\big) = \Big(\hat{\delta}_L(q_L, \omega), \hat{\delta}_M(q_M, \omega)\Big) \in F_L \times F_M$

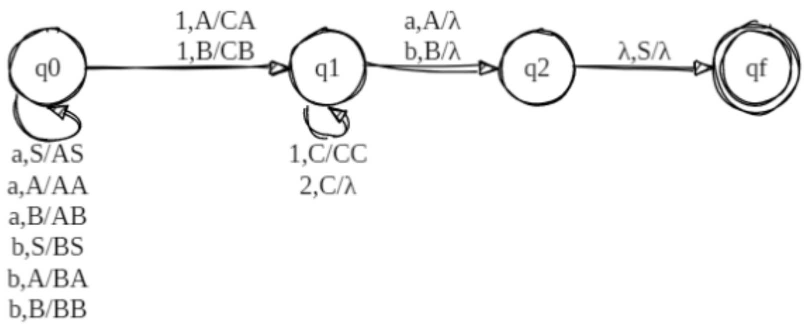
Cosa que ocurre si $\hat{\delta}_L(q_L, \omega) \in F_L \wedge \hat{\delta}_M(q_M, \omega) \in F_M$

Es decir: $\omega \in L(A_L) \wedge \omega \in L(A_M)$

Por lo tanto, $\omega \in L(A) \Leftrightarrow \omega \in L(A_L) \cap L(A_M)$, es decir A acepta $L \cap M$.

3) Un caso especial que no requiere *backtracking* es el ***Análisis Sintáctico Predictivo***.

EL SIGUIENTE AUTÓMATA NO RECONOCE EL LENGUAJE $L = \{\omega \in \{a.b,1,2\}^* | \alpha \in \{a,b\}^+ \wedge \omega = \alpha 1^n 2^n \alpha^r \wedge n > 0\}$ **¿POR QUÉ? ¿QUÉ ESTADOS TIENEN TRANSICIONES INCORRECTAS?**



Porque solo funciona si |alpha| = 1, pues en caso contrario se queda estancado (falta el loop en q2)

EJERCICIO 6: ***GRAMÁTICAS CON ATRIBUTOS.***

EXPLICAR: ¿Qué es un **atributo sintetizado**?

Atributos Sintetizados.

Dada $A \to \alpha$, y sea $A.a = f(\alpha_1.y_1, \alpha_2.y_2, \ldots, \alpha_n.y_n)$ una regla semántica para calcular el atributo a, $A.a$ es un ***atributo sintetizado***, ya que depende del valor de los atributos de los símbolos que están a la derecha de la producción (hijos de A en el árbol sintáctico).

Un ***atributo sintetizado*** para el no terminal A en el nodo N del árbol sintáctico está definido por una regla semántica asociada a la producción en ese nodo en términos de valores de atributos de los hijos de N y del mismo N. El No terminal A aparece en la parte izquierda de la producción.

EJERCICIO 7: *MÁQUINAS DE TURING.*

RELACIONAR EN UNA MISMA FRASE LOS CONCEPTOS:

- **Autómata Linealmente Acotado.**
- **Lenguaje Recursivamente Enumerable.**
- **Lenguaje Sensible al Contexto.**

Sensible al Contexto (tipo 1)	Autómata Linealmente Acotado
Recursivamente Enumerables (tipo 0)	Máquina de Turing

Un Automata Linealmente Acotado es una Maquina de Turing el cual, en vez de ser aceptado por un lenguaje recursivamente enumerable, es aceptado por uno sensible al contexto

Autómatas Acotados Linealmente

Un Autómata Acotado Linealmente es una Máquina de Turing no determinista que usa una cinta de **longitud finita** que **es función lineal de la longitud de la cadena de entrada.**