INFG-TALF Teoría de Autómatas y Lenguajes Formales

Ejercicios de Lenguajes y Gramáticas (T4-P1)

1. Crear una gramática que genere los siguientes lenguajes:
   a) { a, aa, aaa }
   b) { a, aa, aaa, aaaa, aaaaa, …)
   c) { λ, a, aa, aaa }
   d) { λ, a, aa, aaa, aaaa, aaaaa, …)
   La notación empleada para representar cada uno de los lenguajes será:
   a) $\{ a^n \mid n \in [1, 3] \}$
   b) $\{ a^n \mid n > 0 \}$
   c) $\{ a^n \mid n \in [0, 3] \}$
   d) $\{ a^n \mid n \geq 0 \}$

2. Dadas las gramáticas $G=(\Sigma_T, \Sigma_{NT}, S, P_i\}$ donde:

G_1	G_2	G_3	G_4	G_5
$\Sigma_T = \{c\}$ $\Sigma_{NT} = \{S, A\}$	$\Sigma_T = \{c,d\}$ $\Sigma_{NT} = \{S, A\}$	$\Sigma_T = \{c\}$ $\Sigma_{NT} = \{S, A\}$	$\Sigma_T = \{c,d\}$ $\Sigma_{NT} = \{S, A,T\}$	$\Sigma_T = \{c,d\}$ $\Sigma_{NT} = \{S, A\}$
P_1: $S \to \lambda \mid A$ $A \to AA \mid c$	P_2: $S \to \lambda \mid A$ $A \to cAd \mid cd$	P_3: $S \to \lambda \mid A$ $A \to AcA \mid c$	P_4: $S \to cA$ $A \to d \mid cA \mid Td$ $T \to Td \mid d$	P_5: $S \to \lambda \mid A$ $A \to Ad \mid cA \mid c \mid d$

   Determinar el lenguaje asociado a dichas gramáticas.

3. Crear una gramática que genere los siguientes lenguajes:
   a) $\{ a^n b^n \mid n > 0 \}$
   b) $\{ a^n b^m \mid n > 0, 0 < m < n \}$
   c) $\{ a^n b^m \mid n > 0, 0 \leq m < n \}$

4. Determinar el tipo de las siguientes gramáticas en la jerarquía de Chomsky, justificándolo:
   a) G=({a,b}, {A,B,S}, S, P),
      P={S::=aA, A::=bB, A::=aA, A::=a, B::=λ}
   b) G=({a,b,c}, {A,B,C,S}, S, P),
      P={S::=aAb, S::=Ba, S::=λ, aAbC::=aAbB, aAbC::=aabC, BCc::=AaCc, BCc::=BaAbc, C::=Ca, C::=a}
   c) G=({casa, jardin, gato}, {S, CASERON, BOSQUE, TIGRE}, S, P),
      P={ S::=TIGRE jardin, S::=BOSQUE CASERON, BOSQUE::=λ,
      jardin CASERON TIGRE casa::=jardin BOSQUE TIGRE casa,
      gato CASERON BOSQUE::=gato BOSQUE casa TIGRE BOSQUE,
      BOSQUE::=TIGRE casa,BOSQUE::=jardin
      }
   d) G=({x,y}, {C,A,B,S}, S, P),
      P={S::=Cx, S::=Cy, S::=By, S::=Ax, S::=x, S::=y, A::=Ax, A::=Cx, A::=x, B::=By, B::=yA, C::=xA}
   e) G=({a,b,c}, {S,B}, S, P),
      P={S::=abc, S::=aBSc, Ba::=aB, Bb::=bb}

Ejercicios de Lenguajes y Gramáticas (T4-P1) SOLUCIONES

1. Crear una gramática que genere los siguientes lenguajes:

a) **{ a, aa, aaa }**
b) **{ a, aa, aaa, aaaa, aaaaa, …)**
c) **{ λ, a, aa, aaa }**
d) **{ λ, a, aa, aaa, aaaa, aaaaa, …)**

La notación empleada para representar cada uno de los lenguajes será:

a) $\{ a^n \mid n \in [1, 3] \}$
b) $\{ a^n \mid n > 0 \}$
c) $\{ a^n \mid n \in [0, 3] \}$
d) $\{ a^n \mid n \geq 0 \}$

Solución:

a) La gramática sólo debe generar tres palabras y se puede admitir que todas ellas están formadas por el símbolo “a”. Por tanto, una posible solución es:
G=({a}, {S}, S, P} donde:
P={S::=a | aa | aaa}

b) En este caso, las palabras del lenguaje están formadas por una o varias *aes*. Por tanto, una posible solución es:
G=({a}, {S, A}, S, P} donde:
P={S::=A
A::= a | aA}
o simplemente,
G=({a}, {S}, S, P} donde:
P={S::=a | aS}

c) Este caso es similar al planteado en el primer apartado pero, a diferencia de éste, la palabra vacía debe pertenecer al lenguaje. Por tanto, una posible solución es:
G=({a}, {S}, S, P} donde:
P={S::= λ | a | aa | aaa}

d) Este caso es similar al planteado en el segundo apartado pero, a diferencia de éste, la palabra vacía debe pertenecer al lenguaje. Por tanto, una posible solución es:
G=({a}, {S, A}, S, P} donde:
P={S::= λ |A
A:= a | aA}
o simplemente,
G=({a}, {S}, S, P} donde:
P={S::= λ | aS}

2. Dadas las gramáticas G=(Σ_T, Σ_{NT}, S, P_i} donde: (T4p1E2)

G_1	G_2	G_3	G_4	G_5
$\Sigma_T = \{c\}$ $\Sigma_{NT} = \{S, A\}$	$\Sigma_T = \{c,d\}$ $\Sigma_{NT} = \{S, A\}$	$\Sigma_T = \{c\}$ $\Sigma_{NT} = \{S, A\}$	$\Sigma_T = \{c,d\}$ $\Sigma_{NT} = \{S, A,T\}$	$\Sigma_T = \{c,d\}$ $\Sigma_{NT} = \{S, A\}$
P_1: $S\rightarrow\lambda \mid A$ $A\rightarrow AA \mid c$	P_2: $S\rightarrow\lambda \mid A$ $A\rightarrow cAd \mid cd$	P_3: $S\rightarrow\lambda \mid A$ $A\rightarrow AcA \mid c$	P_4: $S\rightarrow cA$ $A\rightarrow d \mid cA \mid Td$ $T\rightarrow Td \mid d$	P_5: $S\rightarrow\lambda \mid A$ $A\rightarrow Ad \mid cA \mid c \mid d$

Determinar el lenguaje asociado a dichas gramáticas.

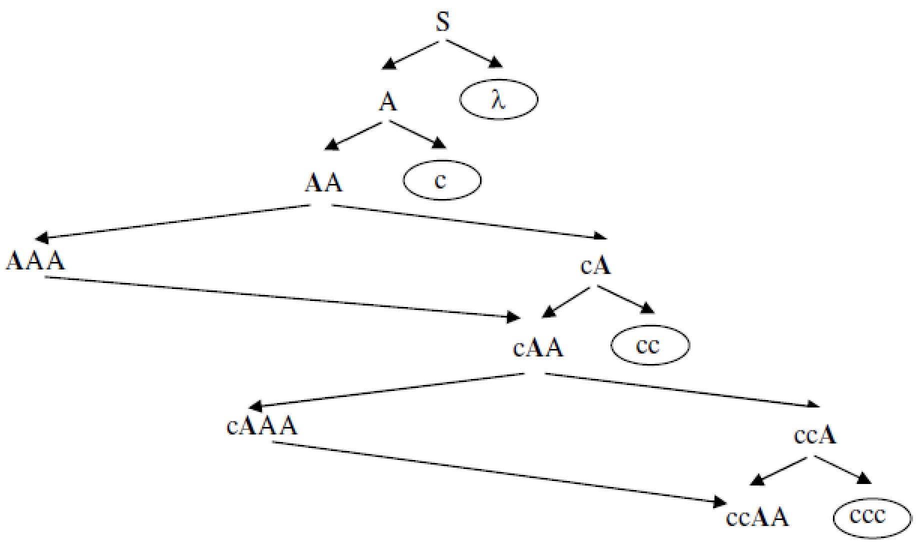
Solución:

El lenguaje asociado a una gramática se corresponde con el conjunto de todas las sentencias de la gramática:

$$L(G) = \{x \,/\, S^*\rightarrow x \;\; \text{AND } x\in\Sigma^* \}$$

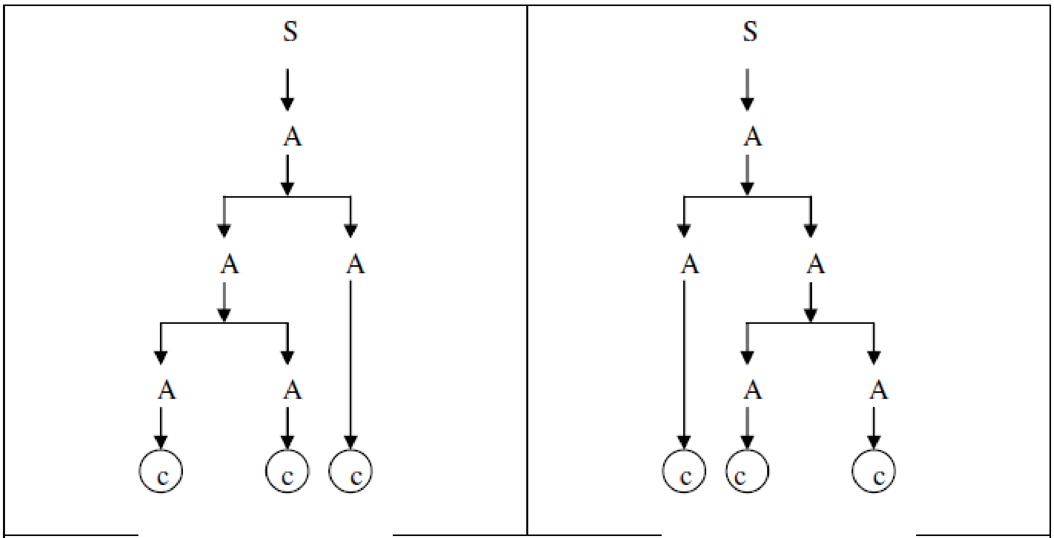
Una forma de encontrar las sentencias (o palabras) del lenguaje asociado a una gramática es apoyarse en una estructura en forma de árbol en la que la raíz es el axioma y las hojas las palabras del lenguaje. Para construir este árbol, hay que tener presente que sus nodos, en general, se corresponden con palabras temporales que derivan en otras palabras temporales. El paso de una a otra se consigue aplicando producciones.
En gramáticas de tipo 2 ó 3 esta transformación se puede simplificar asumiendo que las nuevas palabras temporales se obtienen a partir de su progenitor derivando el símbolo NT situado más a la izquierda.

a) S::=λ | A
A::=AA | c



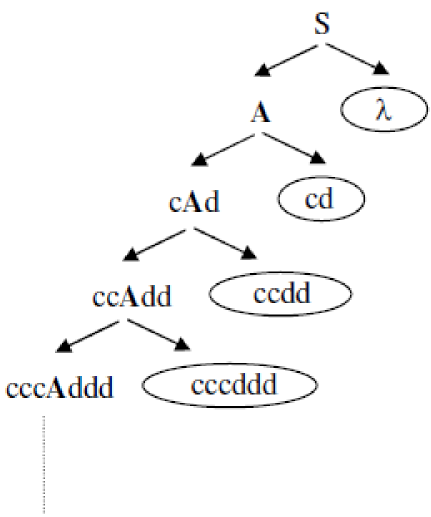
$L(G_1)=\{ \lambda, c, cc, ccc, \ldots \}= \{\lambda, c^n\}$ con n=1, 2, 3, ……

Determinadas palabras se pueden encontrar por más de un camino:
La Gramática es ambigua.



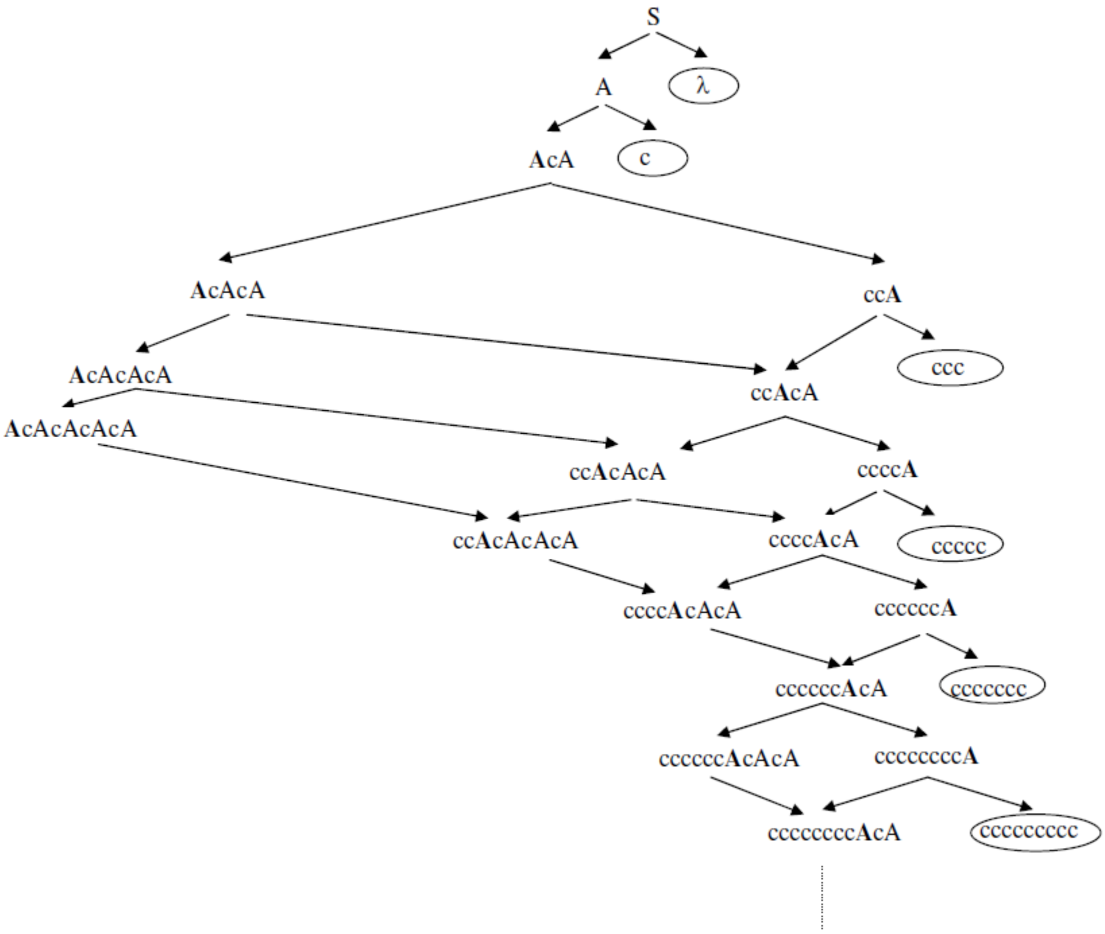
Dos árboles sintácticos (o de derivación) que producen la sentencia ccc

b) S::=λ | A
A::=cAd | cd



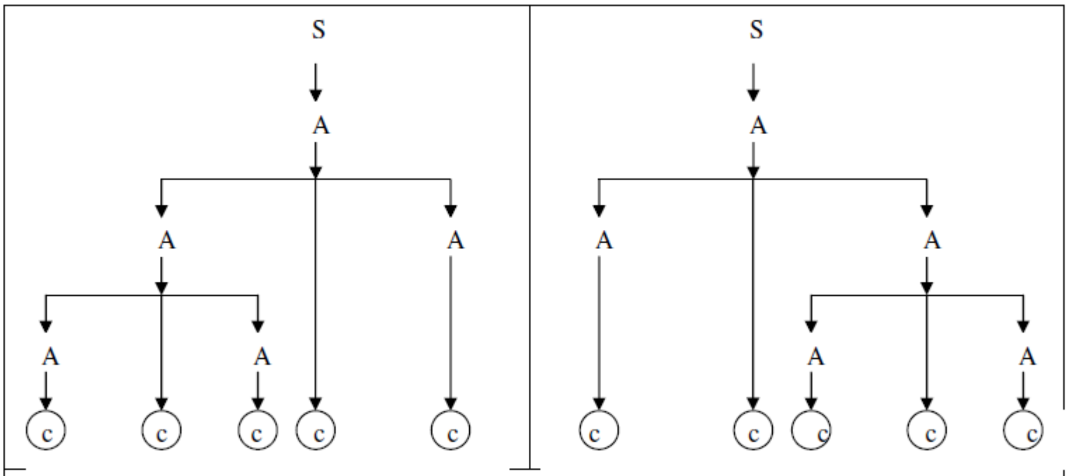
$L(G_2)=\{ \lambda, cd, ccdd, cccddd, \ldots \} = \{\lambda, c^n d^n\}$ con n=1, 2, 3, ……

c) S::=λ | A
A::=AcA | c



L(G_3)={ λ, c, ccc, ccccc, ccccccc,…. }= {λ, c^n} con n=1, 3, 5, ……=
={λ, c^{2n+1}} con n=0,1, 2, 3, ……

Determinadas palabras se pueden encontrar por más de un camino:
La Gramática es ambigua.



Dos árboles sintácticos (o de derivación) que
producen la sentencia ccccc

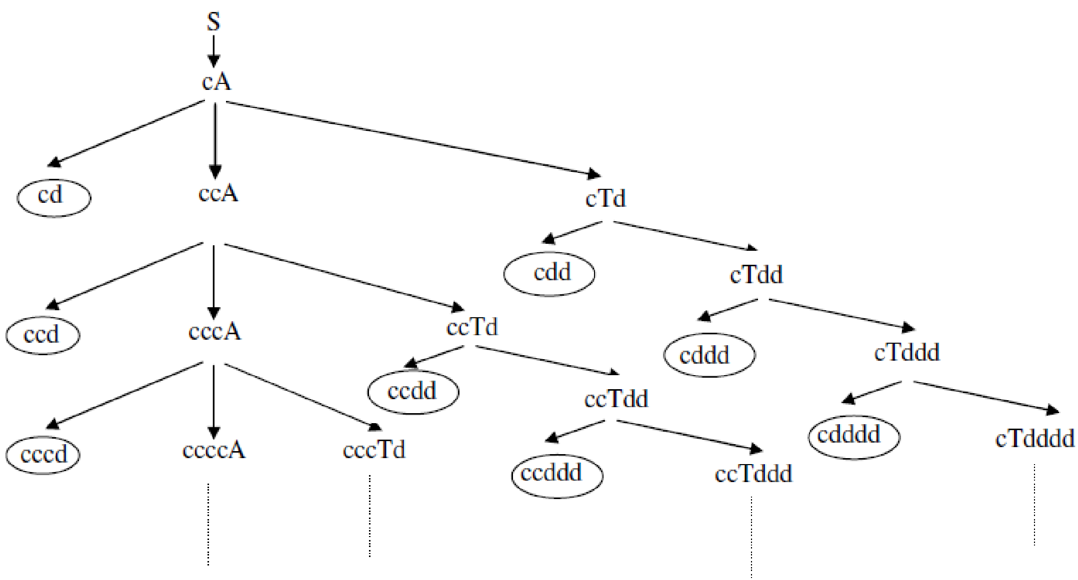
d) S::=cA
A::=d | cA | Td
T::=Td | d

En ocasiones, analizar la semántica de las producciones que definen la gramática puede resultar más factible que intentar encontrar las palabras del lenguaje asociado a una palabra y, a partir de ellas, obtener su definición genérica. Así, por ejemplo, analizando las producciones que definen esta gramática es fácil obtener las siguientes conclusiones:

S → cA: **"la palabra empieza por *c*"**
A→ | cA: "añade ene *ces*"
A→ d | Td: **"la palabra termina en d"**
T → Td | d: "añade eme *des*"

A partir de este análisis es relativamente fácil determinar que las palabras del lenguaje asociado a esta gramática se ajustan a:
$L(G_d)=\{c^n d^m \ / \ n,m\geq 1\}$

Construyendo el árbol, se encuentra que:



$L(G_4)=\{$ cd, ccd, cccd, …..,cdd, ccdd,….. ccddd, ccddd, cdddd, …. $\}= \{c^n d^m\}$ con $n,m\geq 1$

e) S::=λ | A
A::=cd |Ad | cA

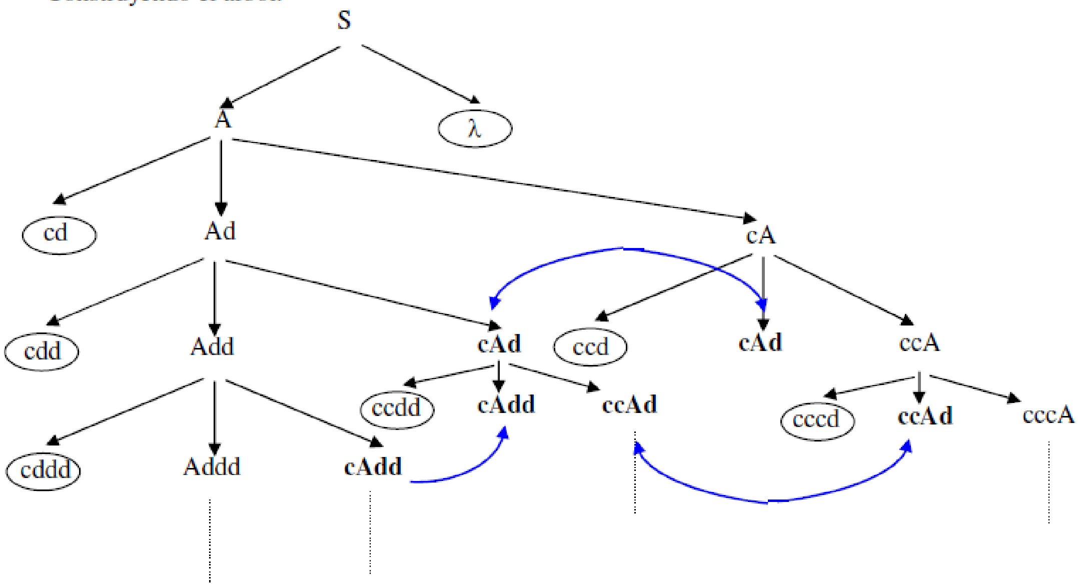
Analizando las producciones se puede concluir:

S::=λ	"El lenguaje contiene la palabra vacía"
S::= A ; A::=cd	"cd es una palabra del lenguaje"
S::= A ; A::=Ad	"Las palabras terminan en una cadena de *des*"
S::= A ; A::=cA	"Las palabras comienzan por una cadena de *ces*"

A partir de este análisis se puede determinar que las palabras del lenguaje asociado a esta gramática se ajustan a:

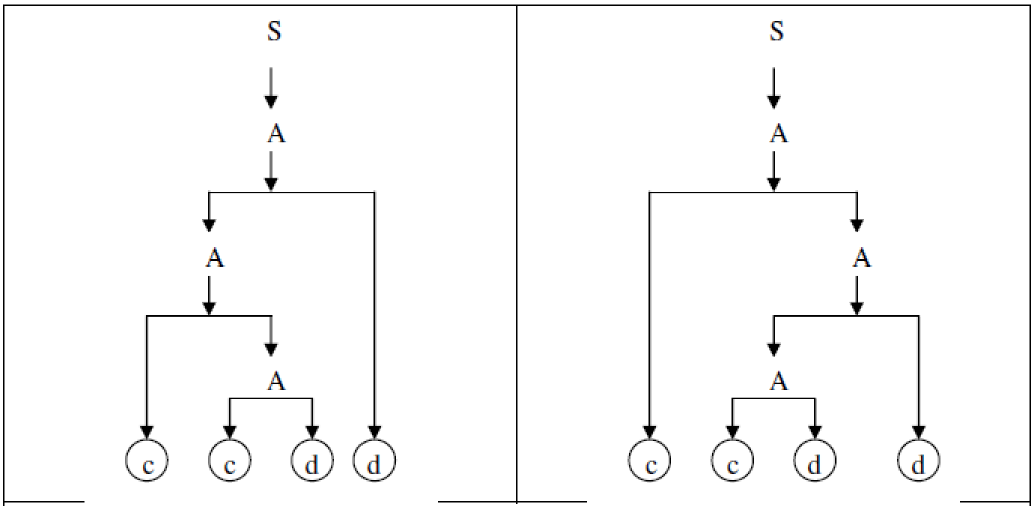
$L(G_e)= \{\lambda, c^n d^m \ / \ n,m \geq 1\}$

Construyendo el árbol:



$L(G_5)=\{ \lambda$, cd, cdd, cddd,...... ccd, ccdd, ccddd,.... $\}= \{\lambda, c^n d^m\}=$ con n,m>=1

Determinadas palabras se pueden encontrar por más de un camino:
La Gramática es ambigua.



Dos árboles sintácticos (o de derivación) que producen la sentencia ccdd

3. Crear una gramática que genere los siguientes lenguajes

a) $\{ a^n b^n \mid n > 0 \}$
b) $\{ a^n b^m \mid n > 0, 0 < m < n \}$
c) $\{ a^n b^m \mid n > 0, 0 \leq m < n \}$

Solución:

a) $\{ a^n b^n \mid n > 0 \}$
Las palabras empiezan por una cadena de *aes* y terminan por una cadena de *bes* existiendo el mismo número de *aes* que de *bes*. Por tanto, cada vez que se añade una *a*, hay que añadir una *b*.
G=({a, b}, {S}, S, P} donde:
P={S::=ab | aSb}

b) $\{ a^n b^m \mid n > 0, 0 < m < n \}$
Las palabras empiezan por una cadena de *aes* y terminan por una cadena de *bes*. Como mínimo tiene que haber una *be* y dos *aes*. Cada vez que se añade una *be* hay que añadir, al menos, una *a*
G=({a, b}, {S}, S, P} donde:
P={S::=aab | aS| aSb}

c) $\{ a^n b^m \mid n > 0, 0 \leq m < n \}$
Las palabras empiezan por una cadena de *aes* y **pueden** terminan en una cadena de *bes*. Como mínimo tiene que haber una *a*. Cada vez que se añade una *be* hay que añadir, al menos, una *a*
G=({a, b}, {S}, S, P} donde:
P={S::=a | aS| aSb}

4. Determinar el tipo de las siguientes gramáticas en la jerarquía de Chomsky, justificándolo:

a) G=({a,b}, {A,B,S}, S, P),
P={S::=aA, A::=bB, A::=aA, A::=a, B::=λ}

b) G=({a,b,c}, {A,B,C,S}, S, P),
P={S::=aAb, S::=Ba, S::=λ, aAbC::=aAbB, aAbC::=aabC, BCc::=AaCc, BCc::=BaAbc, C::=Ca, C::=a}

c) G=({casa, jardin, gato}, {S, CASERON, BOSQUE, TIGRE}, S, P),
P={ S::=TIGRE jardin, S::=BOSQUE CASERON, BOSQUE::=λ,
jardin CASERON TIGRE casa::=jardin BOSQUE TIGRE casa,
gato CASERON BOSQUE::=gato BOSQUE casa TIGRE BOSQUE,
BOSQUE::=TIGRE casa,BOSQUE::=jardin
}

d) G=({x,y}, {C,A,B,S}, S, P),
P={S::=Cx, S::=Cy, S::=By, S::=Ax, S::=x, S::=y, A::=Ax, A::=Cx, A::=x, B::=By, B::=yA, C::=xA}

e) G=({a,b,c}, {S,B}, S, P),
P={S::=abc, S::=aBSc, Ba::=aB, Bb::=bb}

Solución:

a) Tipo 0. Podría ser de tipo 3 pero tiene una regla compresora (B::=λ).

b) Tipo 1. En las producciones 4, 5, 6 y 7 se mantiene el contexto; y el resto serían válidas en una gramática de tipo 2.

c) Tipo 0. Por regla compresora (BOSQUE::=λ). El resto de reglas mantienen el contexto, siendo válidas en una gramática de tipo 1.

d) Tipo 2. Porque hay reglas de producción tipo Nt::=t Nt (G3LD) y tipo Nt::=Nt t (G3LI) mezcladas en la misma gramática.

e) Tipo 0. Porque la regla 3 (Ba::=aB) no mantiene el contexto.