TEMA 1: INTRODUCCIÓN A LA COMPUTACIÓN

DANIEL ALCONCHEL VÁZQUEZ

Un **alfabeto** es un conjunto finito *A*. Sus elementos se llamarán **símbolos o letras**.

```
Notación:
    - Alfabetos: A,B,C...
    - Símbolos: a,b,c...
```

Una **palabra** sobre el alfabeto *A* es una sucesión finita de elementos de *A*.

$$u = a_1 \dots a_n, \quad a_i \in A, \forall i = 1, \dots, n$$

```
Ejemplo:
Si tenemos A={0,1}, entonces 0011 es una palabra del alfabeto
```

El **conjunto de todas las palabras** sobre un alfabeto *A* se nota como *A* *.

```
Notación:
    - Palabras: u,v,x,y...
    - Longitud de la palabra (número de símbolos que la forman): si u=a_1...a_n -
-> |u|=n
    - Palabra vacía: \epsilon (tiene longitud cero)
    - El conjunto de cadenas sobre un alfabeto A, excluyendo la cadena vacía: A^+
```

Llamamos **concatenación** a:

$$u, v \in A^*, u = a_1 \dots a_n;\ v = b_1 \dots b_m \overset{concatenacion}{\Longrightarrow} uv = u.v = a_1 \dots a_n b_1 \dots b_m$$

```
Ejemplo:
Si u=011 y v=1010, entonces tenemos que uv=0111010
```

La concatenación verifica las siguientes propiedades:

$$1)|u.v| = |u| + |v| \qquad \forall u, v \in A^*$$
$$2)Asociativa : u.(v.w) = (u.v).w, \qquad \forall u, v, w \in A^*$$
$$3)Elemento\ neutro : u.\epsilon = \epsilon.u = u, \qquad \forall u \in A^*$$

Definimos **prefijo** y **sufijo** con la connotación del lenguaje natrual. Matemáticamente sería:

$$Prefijo : Si\ u \in A^*, v\ es\ prefijo\ si\ \exists z \in A^* : vz = u$$
$$Sufijo : Si\ u \in A^*, v\ es\ un\ sufijo\ si\ \exists z \in A^* : zv = u$$

Se dicen **propios** si no son ni el vacio ni el total.

Definimos **iteración n-ésima** de una cadena (u^n) como la concatenación con ella misma n veces. Cumple las siguientes propiedades:

$$1)u^0 = \epsilon$$
$$2)u^{i+1} = u^i.u, \qquad \forall i \geq 0$$

```
Ejemplo:
Si u=010, entonces u^3=010010010
```

Definimos **cadena inversa** como invertir de orden los símbolos que componen la cadena. Matemáticamente:

$$u = a_1 ... a_n \in A^* \overset{c.inversa}{\Longrightarrow} u^{-1} = a_n ... a_1 \in A^*$$

```
Ejemplo:
Si u=011, entonces u^-1=110
```

Un **lenguaje** sobre el alfabeto *A* es un subconjunto del conjunto de las cadenas sobre *A*: $A : L \subseteq A^*$.

Un conjunto se dice **numerable** si existe una aplicación inyectiva de este conjunto en el conjunto de los números naturales, o lo que es lo mismo, se le puede asignar un número natural a cada elemento del conjunto de tal manera que dos elementos distintos tengan números distintos.

Aparte de las operaciones de **unión** e **intersección** de lenguajes, también existe la operación **concatenación**, la cual se define como:

$$L_1, L_2 \ son \ lenguajes \ de \ A \overset{concatenacio}{\Longrightarrow} L_1L_2 = \{u_1u_2 : u_1 \in L_1, u_2 \in L_2\}$$

Tiene las siguientes propiedades:

$$1) L\emptyset = \emptyset L = \emptyset$$
$$2) Elemento \ Neutro : \{\epsilon\}L = L\{\epsilon\} = L$$
$$3) Asociativa : L_1(L_2L_3) = (L_1L_2)L_3$$

Ejemplo:

$$L_1 = \{0^i1^i : i \geq 0\}, L_2 = \{1^i0^i : i \geq 0\} \implies L_1L_2 = \{0^i1^i1^j0^j : i, j \geq 0\}$$

La **iteración** de lenguajes de define de forma recursiva:

$$L^0 = \{\epsilon\}, \qquad L^{i+1} = L^iL$$

Si *L* es un lenguaje sobre el alfabeto *A*, se define la **clausura de Kleene** de *L* como:

$$L^* = \bigcup_{i \geq 0} L^i, \qquad L^+ = \bigcup_{i \geq 0} L^i$$

```
Ejemplo:
Si L={0,01}, entonces:
    -L^*= Conjunto de palabras sobre {0,1} en las que 1 siempre va precedido de 0
    -L^+= conjunto de palabras sobre {0,1} en las que 1 siempre va precedido de 0
y distintas de la palabra vacía
```

El **lenguaje inverso** de *L* es el lenguaje dado por:

$$L^-1 = \{u : u^-1 \in L\}$$

La **cabecera** de *L* es el lenguaje dado por:

$$CAB(L) = \{u : u \in A^* y \ \exists v \in A^* \ tal \ que \ uv \in L\}$$

Si A_1 y A_2 son dos alfabetos, una aplicación

$$h : A_1^* \to A_2^*$$

se dice que es un **homomorfismo** si y solo si

$$h(uv) = h(u)h(v)$$

Ejemplo:

$$\begin{gathered}
Si\ A_1\{0,1,2,3,4,5,6,7,8,9\}, A_2 = \{0,1\} \\
h(0) = 0000, \quad h(1) = 0001, \quad h(2) = 0010, \quad h(3) = 0011 \\
h(4) = 0100, \quad h(5) = 0101, \quad h(6) = 0110, \quad h(7) = 0111 \\
h(8) = 1000, \quad h(9) = 1001 \\
h(034) = h(0)h(3)h(4) = 000000110100
\end{gathered}$$

Relación 1

1. Describir el lenguaje generado por la siguiente gramática:

$$S \to XYX$$
$$X \to aX \mid bX \mid \epsilon$$
$$Y \to bbb$$

El lenguaje generado por una gramática $G=(V,T,P,S)$ al conjunto de cadenas formadas por símbolos terminales y que son derivables a partir del símbolo de partida:

Conjunto de todas las palabras sobre un alfabeto

$$L(G) = \{u \in T^* \mid S \overset{*}{\Rightarrow} u\}$$

- V es un alfabeto de símbolos no terminales.
- T es un alfabeto de símbolos terminales.
- P es un conjunto finito de pares (α, β), llamados reglas de producción
- S es un elemento de V, llamado símbolo de partida.

Tenemos que $G=(V=\{S,X,Y\}, T=\{a,b\}, P, S)$. Sabemos que $L(G)=\{p \in T^* : S \overset{*}{\Rightarrow} p\}$, luego $L(G)=\{ubbbw : u,w \in T^*\}$. Otra forma de escribirlo es $L(G)=\{u \in T^* : bbb \text{ es una subcadena de } u\}$

$S \to XYX \longrightarrow \underbrace{X\,bbb\,X}_{aX \mid bX \mid \epsilon} \longrightarrow \{ubbbw : u,w \in T^*\}$

Dem

1) Sea $p \in L(G) \Rightarrow \exists$ una sucesión de pasos de derivación a partir de S, aplicando las reglas de producción obteniendo p:

$$S \Rightarrow XYX \Rightarrow X\,bbb\,X \Rightarrow \underbrace{\{a,b\}^*}_{u}\,bbb\,\underbrace{\{a,b\}^*}_{w}, \quad u,w \in T^*$$

2) Sea $a \in L$, a tiene la siguiente forma:

$$u_1 \dots u_n\, bbb\, w_1 \dots w_m$$

$$S \Rightarrow XYX \Rightarrow \begin{cases} \text{si } u_1 = a \Rightarrow X = aX \\ \text{si } u_1 = b \Rightarrow X = bX \end{cases} \quad \text{sucesivamente}$$

$$Y \Rightarrow bbb$$

Obteniendo así cualquier palabra $ubbbw$.

▷ ¿Es regular la gramática?

Jerarquía de Chomsky:

- Tipo 0: Cualquier gramática. Sin restricciones. Lenguajes recursivamente enumerables.
- Tipo 1: Si todas las producciones tienen la forma

$$\alpha_1 A \alpha_2 \longrightarrow \alpha_1 \beta \alpha_2$$

donde $\alpha_1, \alpha_2, \beta \in (V \cup T)^*$, y $\beta \neq \epsilon$, excepto $S \to \epsilon$. Lenguajes dependientes de contexto.

- Tipo 2: Si cualquier producción tiene la forma

$$A \to \alpha$$

donde $A \in V$, $\alpha \in (V \cup T)^*$.

- Tipo 3: Si toda regla tiene la forma

$$A \to uB \text{ ó } A \to u$$

donde $u \in T^*$ y $A, B \in V$. Conjuntos Regulares.

No es una gramática de tipo 3 porque $S \to XYX$ solo tiene variables, no incluye símbolos terminales.

② Describir el lenguaje generado por la gramática:

$$S \to aX$$
$$X \to aX \mid bX \mid \varepsilon$$

Tenemos $G = (V = \{S, X\}, T = \{a,b\}, P, S)$. Sabemos que $L(G) = \{p \in T^* : S \overset{*}{\Rightarrow} p\}$, luego $L(G) = \{p \in T^* : p \text{ empieza por } a\} = \{au : u \in T^*\}$

$S \to aX \to aaX \mid abX \mid a$ $\qquad S \xrightarrow{a} (X)$ ab

③ Describir el lenguaje generado por la gramática:

$$S \to XaXaX$$
$$X \to aX \mid bX \mid \varepsilon$$

Número de a en la cadena u

Tenemos $G = (V = \{S, X\}, T = \{a,b\}, P, S)$ y como $L(G) = \{p \in T^* : S \overset{*}{\Rightarrow} p\}$, tenemos $L(G) = \{u \in T^* : \boxed{N_a(u)} \geq 2\}$

$S \to XaXaX \to (aX \mid bX \mid \varepsilon)\, a\, (aX \mid bX \mid \varepsilon)\, a\, (aX \mid bX \mid \varepsilon)$

④ Describir el lenguaje generado por la siguiente gramática:

$$S \to SS \mid XaXaX \mid \varepsilon$$
$$X \to bX \mid \varepsilon$$

Tenemos que $G = (V = \{S, X\}, T = \{a,b\}, P, S)$ y sabemos que $L(G) = \{p \in T^* : S \overset{*}{\Rightarrow} p\}$, luego $L(G) = \{u \in T^* : N_a(u) = 0 \bmod 2$ (Número de a par)$\}$

$S \to SS \mid XaXaX \mid \varepsilon$
$\qquad\quad \downarrow$
$\qquad\quad bX \mid \varepsilon$

⑤ Encontrar una gramática libre de contexto que genere el lenguaje sobre el alfabeto $\{a,b\}$ de las palabras que tienen más a que b (al menos una más).

Tenemos $G = (V = \{S, X\}, T = \{a,b\}, P, S)$ y consideremos:

$$S \to XaX$$
$$X \to aX \mid abX \mid baX \mid aXb \mid bXa \mid \varepsilon$$

⑩ Determinar si el lenguaje generado por la gramática

$$S \to XX$$
$$S \to XXX$$
$$X \to aX \mid Xa \mid b$$

es regular. Justificar su respuesta.

Recordemos que un lenguaje es regular si toda regla es de la forma $A \to uB$ o $A \to u$, con $u \in T^*$ y $A, B \in V$. Para facilitar este ejercicio, podemos usar conocimientos del tema 2 (un lenguaje es regular si es aceptado por un autómata finito determinista):

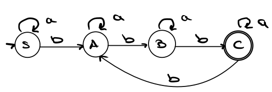
Podemos reescribir las reglas de producción como:

$S \to aS \mid bA$

$A \to aA \mid bB$

$B \to aB \mid bC$

$C \to aC \mid bA$



$\Rightarrow$ Luego, es regular

⑪ Dado un lenguaje L sobre un alfabeto A, ¿es L^* siempre numerable? ¿nunca lo es? ¿o puede serlo unas veces sí y otras, no? Pon ejemplos en este último caso.

Por un lado, tenemos $L^i \subseteq A^*$. Sabemos que A^* es numerable $\Rightarrow L^i$ es numerable.

Queremos probar que $L^* = \bigcup_{i \geq 0} L^i$ es numerable. Sabemos que L^i se define recursivamente como $\begin{cases} L^0 = \{\varepsilon\} \neq \emptyset \\ L^n = L^{n-1}L \end{cases}$. Basta con tomar una aplicación inyectiva continua de la siguiente forma:

$$\bigcup L^i \xrightarrow{g} \underset{(n,m)}{\mathbb{N} \times \mathbb{N}} \xrightarrow{h} \mathbb{N}$$
$$u \longmapsto (i, f_i(u))$$

(12) Dada la gramática $G = (\{S, A\}, \{a, b\}, P, S)$, donde $P = \{S \to abAS,\ abA \to baab,\ S \to a,\ A \to b\}$. Determinar el lenguaje que genera.

$S \to a \Rightarrow$ acaba en a

$S \to abAS \to baabS \longrightarrow baaba$

$S \to abAS \to abbS \longrightarrow abba$

$\Rightarrow$ Cualquier combinación de las cadenas $\begin{cases} abb \\ baab \end{cases}$ combinadas con a al final

$(abb + baab)^* a$

(14) Sea la gramática $G = (\{A, S\}, \{a, b\}, S, P)$, donde las reglas de producción son:

$S \to aS$

$S \to aA$

$A \to bA$

$A \to b$

Determinar el lenguaje generado por la gramática.

El lenguaje generado es $L(G) = \{a^i b^j : i \geq 1, j \geq 1\}$

(15) Dado un lenguaje L sobre un alfabeto A, caracterizar cuando $L^* = L$. Esto es, dar un conjunto de propiedades de L de manera que se cumplen si, y sólo si, $L^* = L$.

Queremos ver que si $L^* = \bigcup_{i \geq 0} L^i = L$ si, y solo si, se cumplen los siguientes propiedades:

P1) $\varepsilon \in L$ $\quad (L^* = L^0 \cup L^+ = \{\varepsilon\} \cup L,\ \varepsilon \in L^*)$

P2) Si $u \in L \Rightarrow uv \in L,\ \forall v \in L$

Vamos a demostrarlo:

$\underline{\Rightarrow)}$ $(L = L^* \Rightarrow P1 \wedge P2)$

- Como $L = L^* \Rightarrow \varepsilon \in L$
- Si $u, v \in L$, $uv \in L^2 \subset L^* = L \Rightarrow uv \in L \Leftrightarrow L = LL$

$\underline{\Leftarrow)}$ $(P1 \wedge P2 \Rightarrow L^* = L)$

Si suponemos que $L^n \subseteq L$

- $L^{n+1} \subseteq L \Leftrightarrow L^{n+1} = L^n \cdot L \subseteq LL = L^2 \subseteq L \Leftarrow$

  $L^{n+1} \subseteq L$, $\bigcup_{i \geq 0} L^i = L^* \subseteq L$

- $\varepsilon \in L \Rightarrow L^0 \in LL^* = L^0 \cup L^+ \subseteq L$

(16) Dados dos homomorfismos $f: A^* \longrightarrow B^*$, $g: A^* \longrightarrow B^*$, se dicen que son iguales si $f(x) = g(x)$, $\forall x \in A^*$. ¿Existe un procedimiento algorítmico para comprobar si dos homomorfismos son iguales?

Tiene que cumplirse:

- $f(u_1, u_2) = f(u_1) f(u_2)$
- $f(\varepsilon) = \varepsilon$

La idea consiste en comprobar el alfabeto (como base):

Si $\underbrace{f(a_i) = g(a_i)}$ $\forall i \Rightarrow f = g$

$\Downarrow$

$f(a_i) = f(a_{i_1}, \ldots, fa_{i_n}) = f(a_{i_1}) \cdots f(a_{i_n})$

(18) Demuestra que, para todo alfabeto A, el conjunto de los lenguajes finitos sobre dicho alfabeto es numerable.

Sea $A = \{a_1, \ldots, a_f\}$ un alfabeto y sea $\mathcal{L}_f$, el conjunto de los lenguajes finitos, $L \in \mathcal{L}_f$, $L = \{w_1, \ldots, w_n\}$. Sea $*$ un carácter tal que $* \notin A$, de forma que $* w_1 * \cdots * w_n$. A cada $a_i \in A$ y a cada $*$ le asignamos $0, \ldots, f-1, f$, de forma que a cualquier $L \in \mathcal{L}_f$ le podemos asignar un número base $|A|+1$.

De esta forma podemos definir una función inyectiva $f: \mathcal{L}_f \to \mathbb{N}$ que asigne a cada lenguaje dicho valor (falta probar que, efectivamente, es inyectiva).

Relación 1-Bis

Sencillos

a) $\{u \in \{0,1\}^* : |u| < 4\}$

$S \to 0A \mid 1A \mid \varepsilon$

$A \to 0B \mid 1B \mid \varepsilon$

$B \to 0 \mid 1 \mid \varepsilon$

b) Palabras con 0's y 1's que no contengan dos 1's consecutivos y que empiecen por un 1 y que terminen por dos 0's.

$S \to 1A$

$A \to 0A \mid 01A \mid 00$

c) El conjunto vacío

$S \to aS$ (Como nunca llega a un símbolo terminal, no puede generar palabras $\Rightarrow \phi$)

e) $\{a^n \in \{a,b\}^*$ con $n \geq 0\} \cup \{a^n b^n \in \{a,b\}^*$ con $n \geq 0\}$

$\underbrace{a \dots a}_{n}$ ó $\underbrace{a \dots a}_{n}\underbrace{b \dots b}_{n}$

$S \to S1 \mid S2$ → Es muy conveniente hacer la unión de lenguajes de esta forma.

$S1 \to aA \mid \varepsilon$

$S2 \to aS2b \mid \varepsilon$

f) $\{a^n b^{2n} c^m \in \{a,b,c\}^*$ con $n,m > 0\}$

$S \to S1S2$

$S1 \to aS1bb \mid abb$ ·) $S1S2 \to aS1bbS2 \to \underbrace{aa}_{n}\underbrace{bbbb}_{2n}S2 \to \underbrace{aa}_{n}\underbrace{bbbb}_{2n}\underbrace{c}_{m}$

$S2 \to cC \mid c$

g) $\{a^n b^m a^n \in \{a,b\}^*$, con $m,n \geq 0\}$

$S \to aSa \mid B \mid \varepsilon$ ·) $aSa \to aaSaa \to aaBaa \to aabBaa \to \underbrace{aa}_{n}\overset{m}{b}\underbrace{aa}_{n}$

$B \to bB \mid \varepsilon$

h) Palabras con 0's y 1's que contengan la subcadena 00 y 11

i) Palíndromos formados con las letras a y b.

Recuerda: un palíndromo es una palabra que se lee igual en un sentido que otro.

$S \to aSa \mid bSb \mid a \mid b \mid \varepsilon$

Media

a) $\{uv \in \{0,1\}^* : u^{-1}$ es prefijo de $v\}$

$S \to S1S2$

$S1 \to 0S10 \mid 1S11 \mid \varepsilon$

$S2 \to 0S2 \mid 1S2 \mid \varepsilon$

b) $\{ucv \in \{a,b,c\}^* : |u| = |v|\}$

$S \to S1cS2$ $XcY \to c$

$S1 \to aS1X \mid bS1X \mid cS1X \mid \varepsilon$ $Xc \to ca \mid cb \mid cc$

$S2 \to YS2a \mid YS2b \mid YS2c \mid \varepsilon$ $cY \to ac \mid bc \mid cc$

c) $\{u1^n \in \{0,1\}^*$ donde $|u| = n\}$

$S \to 0S1 \mid 1S1 \mid \varepsilon$

d) $\{a^n b^n a^{n+1} \in \{a,b\}^*$ con $n \geq 0\}$

$S \to S1S2$ $Hb \to bH$ $S2 \to a$ ·) $aabHbHS2 \to aabbHHS2 \to aabbS2aa \to aabbaaa$

$S1 \to aSbH \mid \varepsilon$ $HS2 \to S2a$

Ejercicios Examen

① $\{a^n b^m c^k : |n-m| = k \text{ con } n, m, k \geq 0\}$

Si $N_a > N_b \Rightarrow n > m \Rightarrow n - m = k \Rightarrow n = k + m$. En caso contrario, $-n+m = k \Rightarrow m = n+k$

$$S \rightarrow S1 \mid S2$$

$S1 \rightarrow bS1A \mid C$ 　　　 $S2 \rightarrow aSB \mid C$

$C \rightarrow cCA \mid \varepsilon$ 　　　 $C \rightarrow cCB \mid \varepsilon$

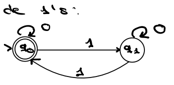
$bA \rightarrow Ab$ 　 $cA \rightarrow Ac$ 　　　 $cB \rightarrow Bc$

$Ab \rightarrow ab$ 　 $Aa \rightarrow aA$ 　　　 $Bc \rightarrow bc$ 　 $Bb \rightarrow bB$

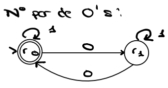
② Palabras con un número par de 0 y 1.

Pensándolo con el autómata:

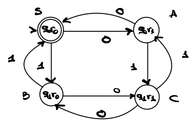
Nº par de 1's:



Nº par de 0's:



Juntándolos:



$S \rightarrow 0A \mid 1B \mid \varepsilon$

$A \rightarrow 0S \mid 1C$

$B \rightarrow 1S \mid 0C$

$C \rightarrow 1A \mid 0B$

③ Encontrar una gramática que genere $\{a^n b^m \in \{a,b\}^* : 3m \geq n \geq 2m\}$

La condición $3m \geq n \geq 2m$ indica que el número de a esté comprendido entre el doble y el triple que el número de b. Sea $G = (V, T, P, S)$, con $V = \{S\}$, $T = \{a, b\}$, y P compuesto por:

$$S \rightarrow aaSb \mid aaaSb \mid \varepsilon$$

Explicación:

- Con aaSb tienes $n = 2m$
- Con aaaSb tienen $n = 3m$
- Intercalándolas tendrás que $3m \geq n \geq 2m$
- Admitimos la palabra vacío, ya que no dice que $n > 0$.

④ Encontrar una gramática que genere $\{a^n b^m c^k : k = 2n + 3m\}$

- $N(c) = 2N(a) + 3N(b)$

Basicamente, consiste en que cada vez que añades una a añade $2c$, de esa forma, tendras que $N(c) = 2N(a)$. Por otro lado, cada vez que añades una b añades $3c$, de esa forma $N(c) = 3N(b)$, luego:

$$S \rightarrow aScc \mid B \mid \varepsilon$$

$$B \rightarrow bBccc \mid \varepsilon$$

$S \rightarrow aScc \rightarrow aaScccc \rightarrow aaBcccc \rightarrow aabBccccccc \rightarrow aab\underbrace{cccc}_{2N(a)}\underbrace{ccc}_{3N(b)}$

⑤ Encuentra una gramática que genere $\{a^n b^m c^k : n = m \text{ ó } m = k\}$

$$S \rightarrow S_1 \mid S_2 \mid \varepsilon$$

$$S_1 \rightarrow XC, \quad X \rightarrow aXb \mid \varepsilon, \quad C \rightarrow cC \mid \varepsilon$$

$$S_2 \rightarrow AY, \quad Y \rightarrow bYc \mid \varepsilon, \quad A \rightarrow aA \mid \varepsilon$$

- $S \rightarrow S_1 \rightarrow XC \rightarrow aXbC \rightarrow aaXbbC \rightarrow \underbrace{aa}_{n}\underbrace{bb}_{m}ccc\ldots$
- $S \rightarrow S_2 \rightarrow AY \rightarrow aa\cdots bYc \rightarrow aa\cdots bbYcc \rightarrow aa\cdots \underbrace{bb}_{m}\underbrace{cc}_{k}$

⑥ Dar una gramática para $\{u0v : u^{-1} \text{ es prefijo de } v\}$, $u, v \in \{a, b, 0\}^*$

$$S \rightarrow S_1 S_2 \mid 0$$

$$S_1 \rightarrow aS_1a \mid bS_1b \mid 0$$

$$S_2 \rightarrow aS_2 \mid bS_2 \mid 0S_2 \mid \varepsilon$$

$S \rightarrow S_1S_2 \rightarrow aS_1aS_2 \rightarrow aaS_1aaS_2 \rightarrow aabS_1baaS_2 \rightarrow \underbrace{aab}_{u}0\underbrace{baa}_{u^{-1}}S_2$ Genera cualquier palabra

7. Encontrar una gramática que genere el lenguaje $\{a^m b^n : m > n\}$

$$S \to a \mid aSb \mid aA$$

$$A \to aA \mid \varepsilon$$

- $S \to a$ ✓
- $S \to aSb \to \underbrace{aa}_{n} S \underbrace{bb}_{m} \ (n=m) \to \underbrace{aaa}_{n} \underbrace{bb}_{m} \ (n>m)$
- $S \to aSb \to \underbrace{aa}_{n} S \underbrace{bb}_{m} \ (n=m) \to \underbrace{aaaA}_{n} \underbrace{bb}_{m}$ $(n>m)$ ya que A genera tantas a como queramos.

8. Encontrar una gramática que genere $L = \{u : N_b(u) \neq N_c(u)\}$, $u \in \{a,b,c\}^*$

$$S \to aS \mid bSX \mid cSY \mid a \mid bX \mid cY$$

$$XY \to H \quad YX \to H$$

$$H \to b \mid c$$

$$XH \to \varepsilon,\ YH \to \varepsilon,\ HX \to \varepsilon,\ HY \to \varepsilon$$

$$\boxed{X \to \varepsilon,\ Y \to \varepsilon}$$

- $aS \to abSX \to abcSYX \to abcaYX \to abcaH \to abcab$

  como está sola es porque $N_b = N_c$

- $aS \to abSX \to abbcbXYXX \to abbcbHXX \to abbcbX \to abbcb$