

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE CIENCIAS



TAREA 01

Gramáticas y árboles de derivación

PRESENTA

Valeria Camacho Hernández - 322007273

ASIGNATURA

Estructuras Discretas 2025-2

PROFESOR

Ulises Rodríguez Domínguez

AYUDANTE

Irvin Javier Cruz González

FECHA

Miércoles 19 de febrero del 2025

Tarea 01

1. Colores hexadecimales. Un color hexadecimal, como los usados en páginas web, tiene el formato #AAAAAA, donde cada A puede tomar un valor del alfabeto 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F.

Define una gramática para producir uno o varios colores hexadecimales separados por una coma. Después de definir la gramática dibuja los árboles de derivación para dos ejemplos distintos (con al menos dos colores cada uno).

Respuesta:

- Reglas de producción (P):

$S := E$

$E := H \text{ var}$

$A := 0 \mid 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9 \mid A \mid B \mid C \mid D \mid E \mid F$

$\text{var} := A \text{ var} \mid A M$

$M := ,$

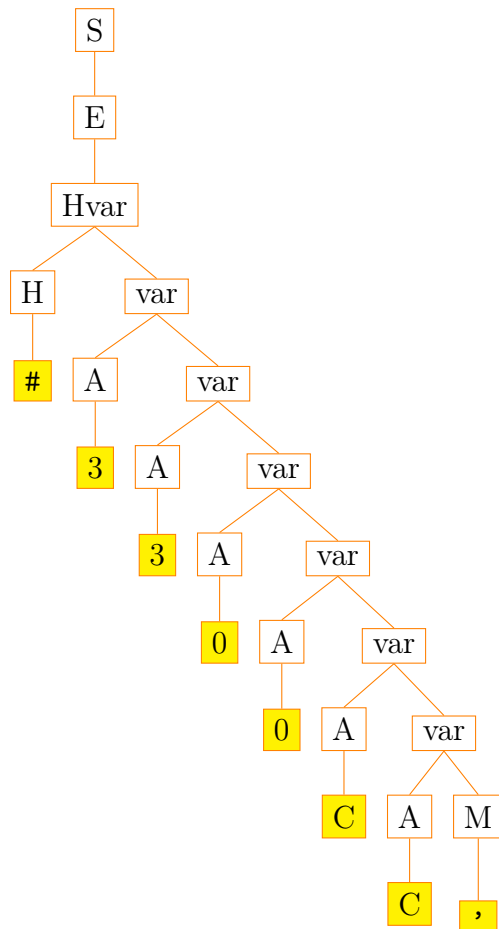
$H := \#$

- Gramática:

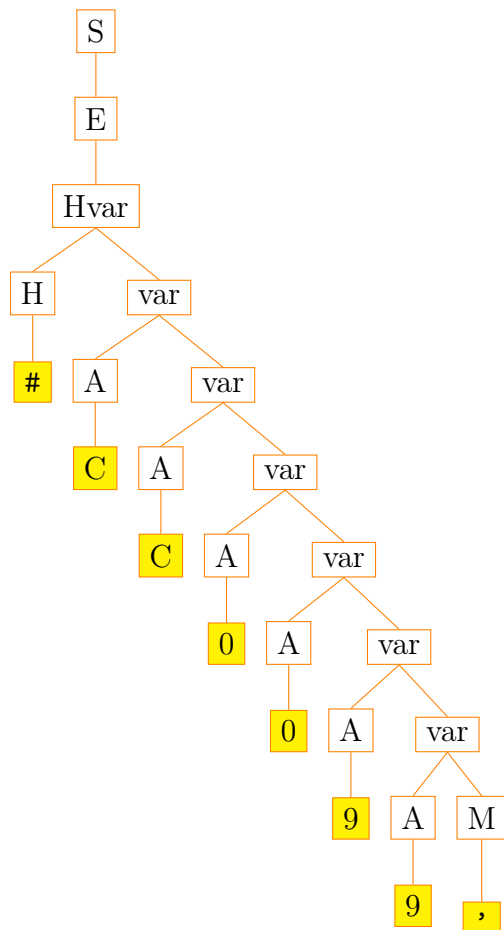
$G(\text{Hex}) = (\{0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F, , \#, \}, \{S,E,\text{var},A,M,H\}, S, P)$

- Ejemplos:

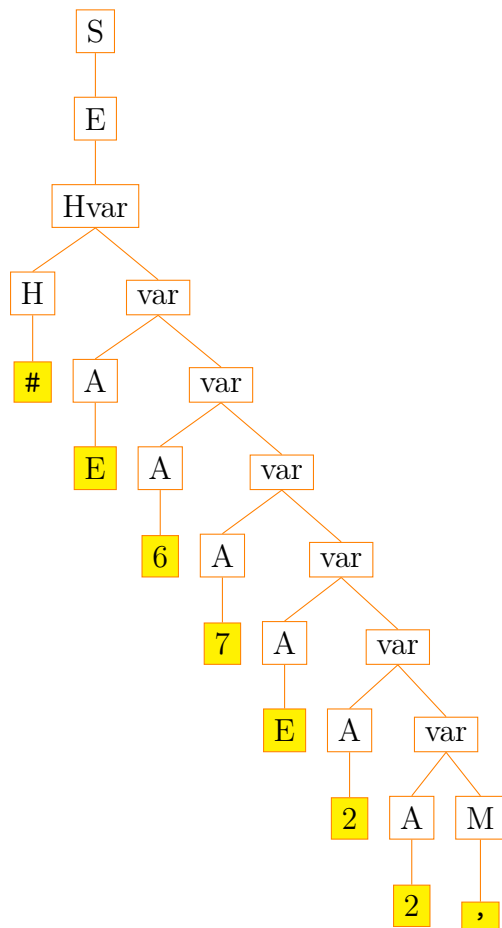
1) # 3300CC,



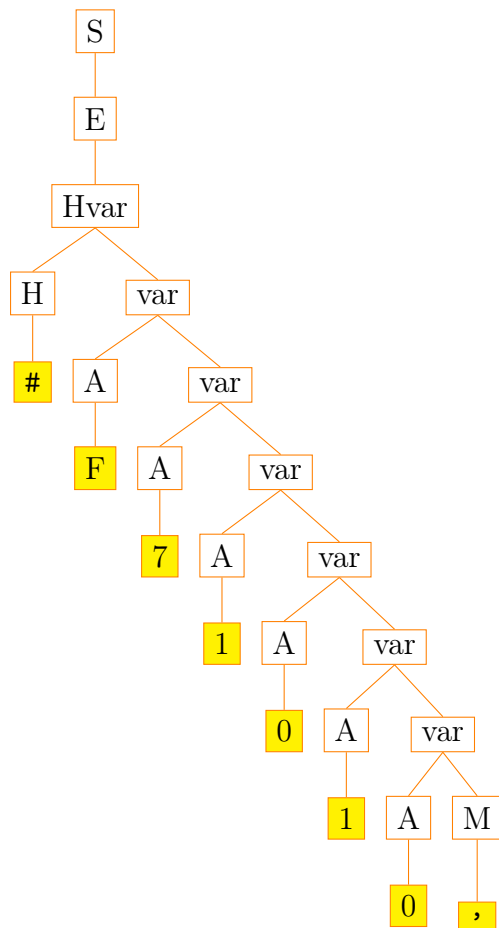
2) #CC0099,



3) # E67E22,



4) # F71010,



2. Como parte de un proceso de automatización del diseño de moléculas en base a propiedades químicas particulares [1], se usa una gramática de libre contexto para generar cadenas de caracteres con el formato SMILES. SMILES [2] es un formato común para describir moléculas con una cadena de caracteres. En éste formato, se asume que los átomos consecutivos en la cadena están conectados mediante un sólo enlace y los enlaces dobles se denotan por =. Por ejemplo CO₂ se representa en una cadena como O = C = O. Las siguientes reglas de producción son un subconjunto de las reglas de producción para la gramática en [1], la cual respeta el formato SMILES:

smiles ::= initial_valence_2 double_bond (1)

initial_valence_2 ::= 'O' | 'S' (2)

double_bond ::= '=' valence_2 (3)

double_bond ::= '=' valence_4 double_bond (4)

valence_2 ::= 'O' | 'S' (5)

valence_2 ::= valence_3 (6)

valence_3 ::= 'N' (7)

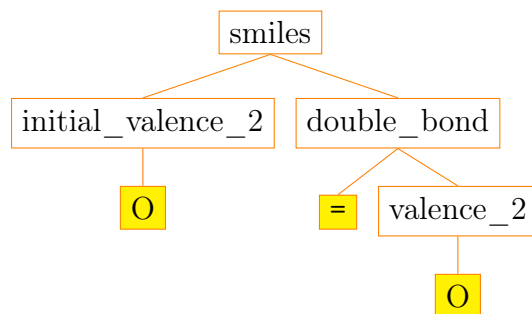
valence_3 ::= valence_4 (8)

valence_4 ::= 'C' (9)

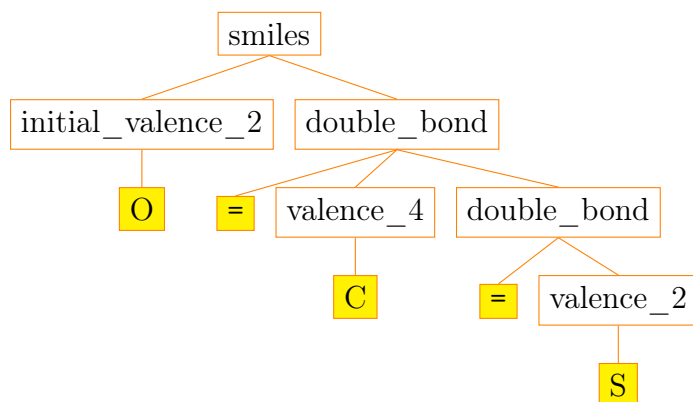
Haz el árbol de derivación para cinco moléculas distintas que se puedan producir con éstas reglas de producción. Además, investiga el nombre real de cada molécula en tus ejemplos y etiqueta el árbol correspondiente.

Respuesta:

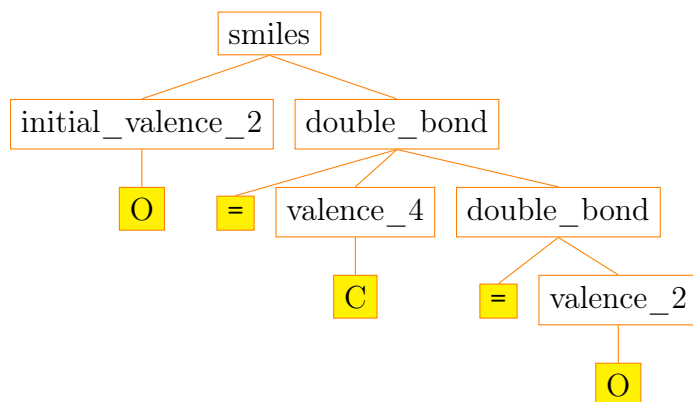
1) O = O es oxígeno



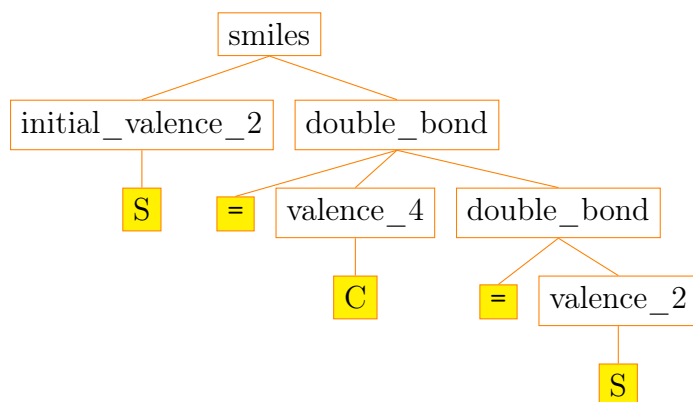
2) $O = C = S$ es sulfuro de carbonillo



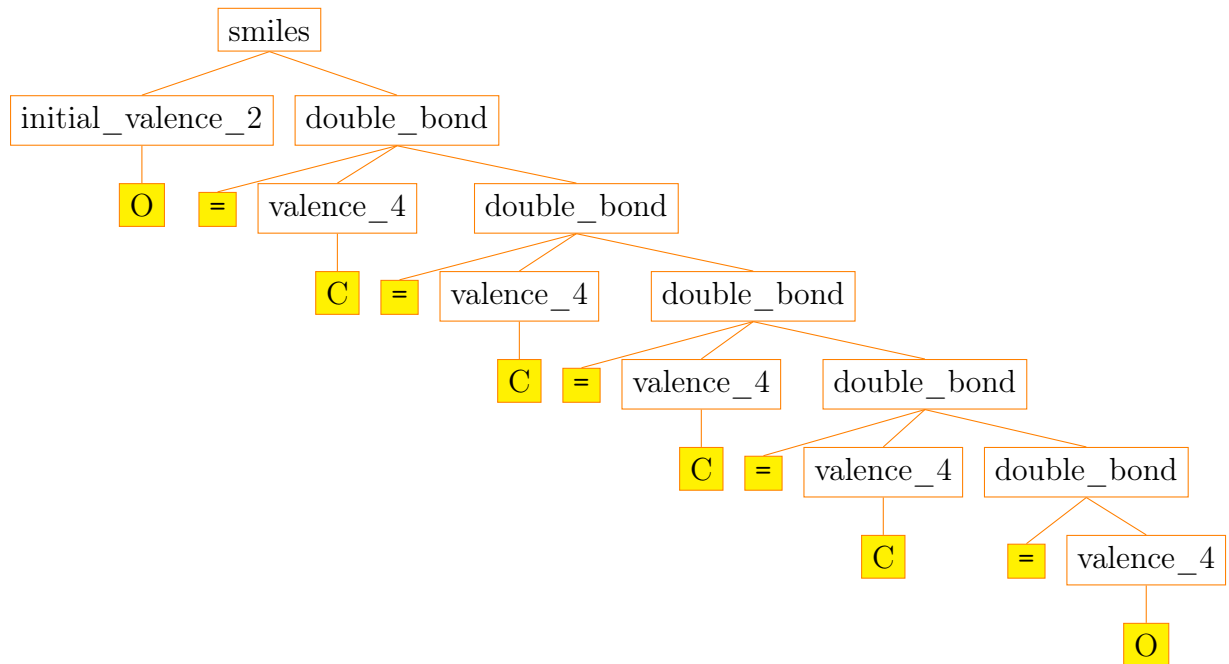
3) $O = C = O$ es dióxido de carbono



4) $S = C = S$ es sulfuro de carbono



5) $O = C = C = C = C = O$ es dióxido de tetracarbono



3. Define una gramática para declarar variables de tipo entero (int) y variables de tipo flotante (float).

Debes considerar lo siguiente:

- (a) Una declaración de variables tipo entero tiene el siguiente formato: `int V;`
donde V puede ser una variable o una lista de variables separadas por una coma.
- (b) Una declaración de variables tipo flotante tiene el siguiente formato: `float V;`
donde V puede ser una variable o una lista de variables separadas por una coma.
- (c) Se tienen cuatro posibles casos, que pueden ser: declarar variables enteras, declarar variables flotantes, declarar variables enteras seguidas de otra declaración de variables flotantes, declarar variables flotantes seguidas de otra declaración de variables enteras. Ésto es:
 - i. `int V;` ii. `float V;` iii. `int V; float V;` iv. `float V; int V;`
- (d) No es necesario que te preocupes por variables repetidas en tus declaraciones.

En un compilador real, una opción posible es aplicar un postprocesamiento a las cadenas de caracteres para detectar variables repetidas.

Después de definir la gramática dibuja los árboles de derivación para dos ejemplos de declaraciones (considerando al menos un ejemplo con combinación de enteros y flotantes).

Respuesta:

- Reglas de producción (P):

$S := E$

$E := U \mid U E \mid D \mid D E$

$U := I V P \mid F V P$

$D := I V P F V P \mid F V P I V P$

$I := \text{int}$

$F := \text{float}$

$V := \text{var } M V \mid \text{var}$

$\text{var} := a \mid b \mid \dots \mid z$

$P := ;$

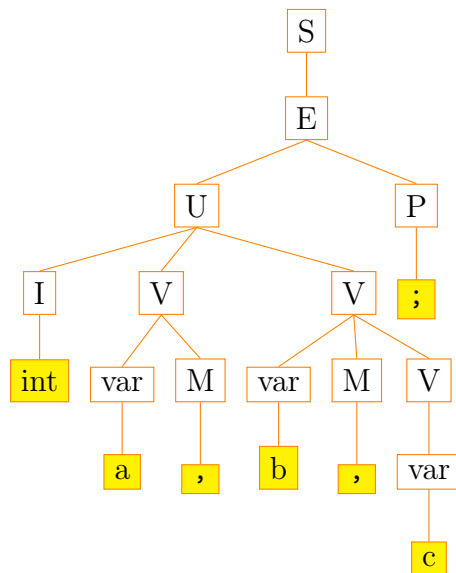
$M := ,$

- Gramática:

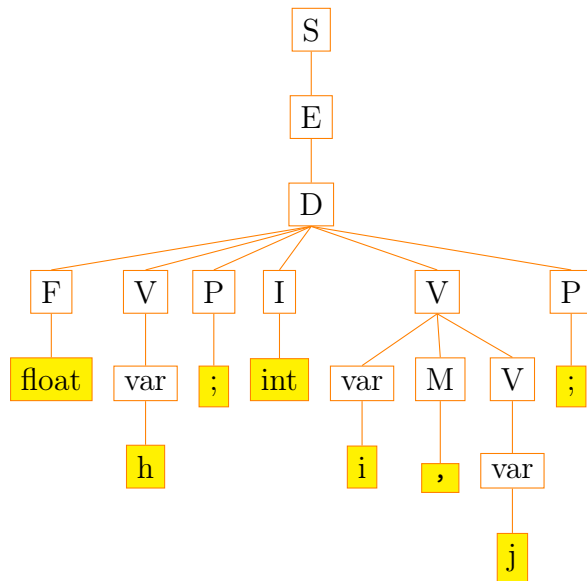
$G(\text{Hex}) = (\{a, b, \dots, z, ,, ;\}, \{S, E, \text{var}, U, D, I, F, V, P, M\}, S, P)$

- Ejemplos:

1) `int a,b,c;`



2) float h; int i,j;



4. ¿En que fallaste la primera vez que cursaste la materia? ¿Como lo piensas superar en este nuevo curso?

Respuesta: Fallé en no preguntar las dudas que tenía a los ayudantes o profesor, y en no ser consistente con estudiar, además de no organizarme con tiempo para las tareas y exámenes. Lo pienso superar acercarme más a los ayudantes cuando tenga dudas o no entienda algo.