WorkShop No. 1 — The Old Times

UNIVERSIDAD DISTRITAL
FRANCISCO JOSÉ DE CALDAS

Members:

Andrés Felipe Salazar Malagón – 20202020043

Teacher:

Carlos Andrés Sierra Virguez

Universidad Distrital Francisco José de Caldas

Faculty of Engineering

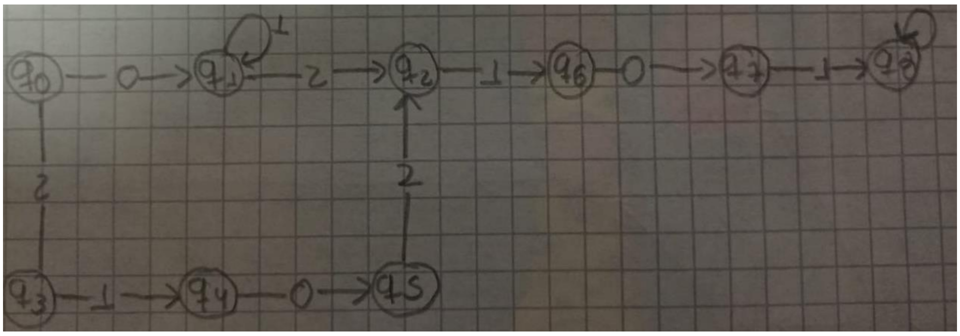
Computer Science III

Bogotá, 2024

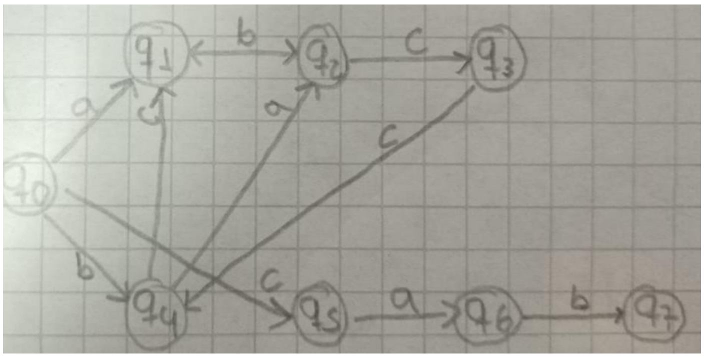
Exercices:

1. For each of the following languages, define the corresponding finite-state machine:

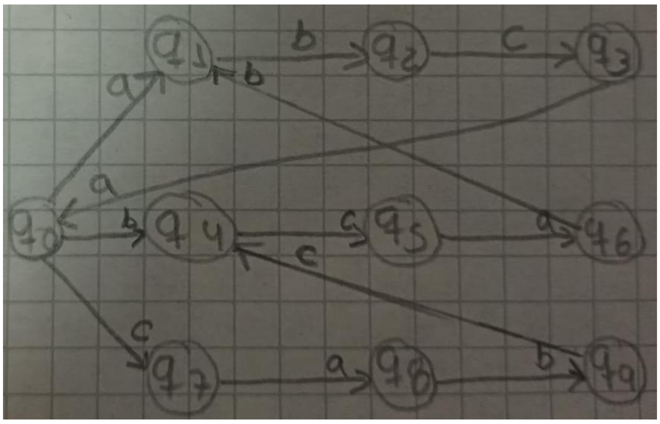
(i) $\Sigma = \{0, 1, 2\}$. $L = (01^*2 \cup 2102)^*101(01 \cup 12 \cup 20)^*$.



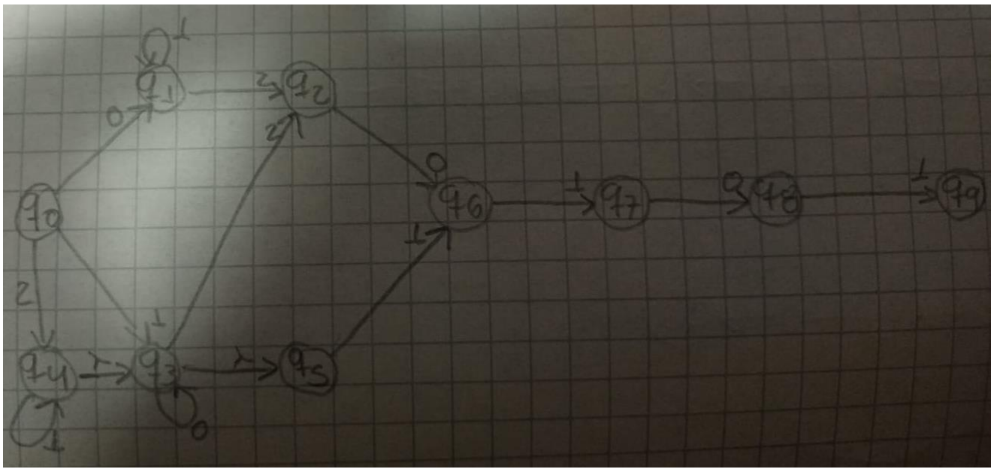
(ii) $\Sigma = \{a, b, c\}$. $L = (abc \cup bca \cup cab)(abc \cup bca \cup cab)^*$.



(iii) $\Sigma = \{a, b, c\}$. $L = (abc \cup bca \cup cab)^*(abc \cup bca \cup cab)$.

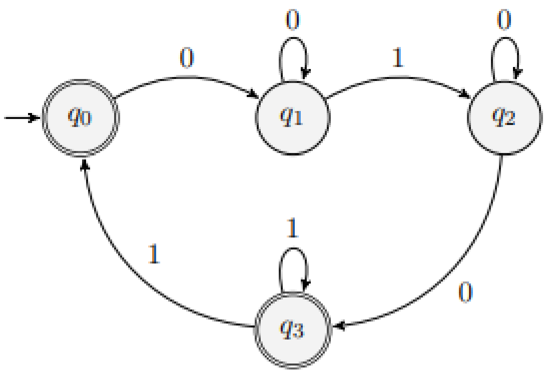


(iv) Σ = {0, 1, 2}. L = (01∗2 ∪ 10∗2 ∪ 21∗0)∗ (01 ∪ 12 ∪ 20)∗101.



2. For each one of the following finite-state machines, define the corresponding regular expression and a generative grammar:

(i) $\Sigma = \{0, 1\}$.



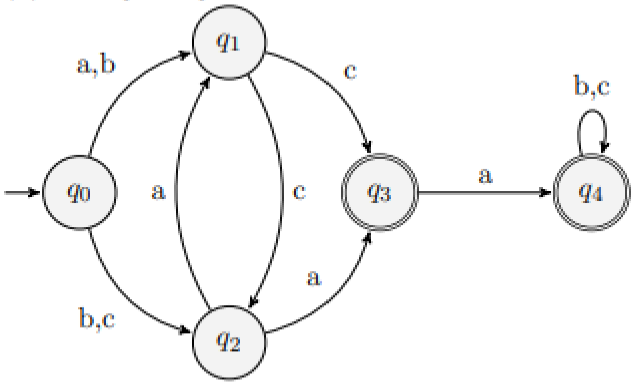
(i) For $\Sigma = \{0,1\}$.

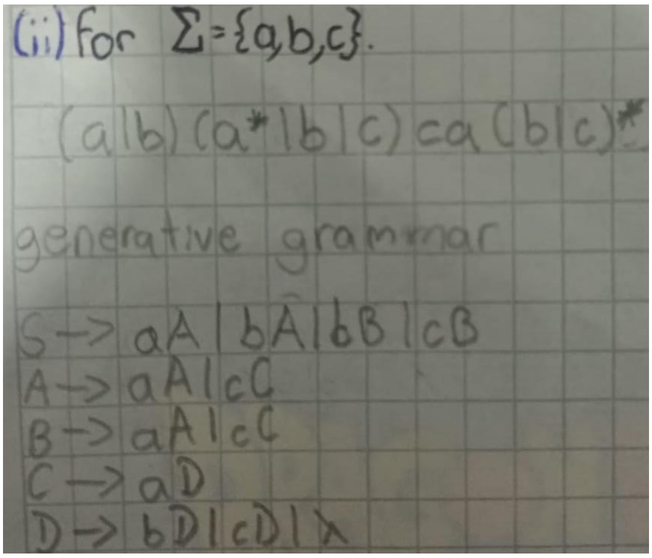
$(0(0^*1)^*0)^*$

Generative grammar

S → 0A | λ
A → 0A | 1B | λ
B → 0B | 1C | λ
C → 1C | 0S | λ

(ii) $\Sigma = \{a, b, c\}$.




(ii) for $\Sigma = \{a,b,c\}$.

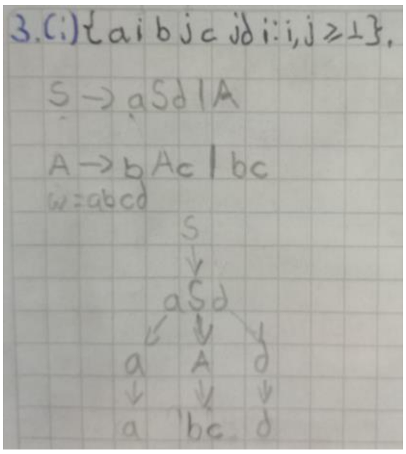
$(a|b)(a^*|b|c)ca(b|c)^*$

generative grammar

$S \to aA \mid bA \mid bB \mid cB$
$A \to aA \mid cC$
$B \to aA \mid cC$
$C \to aD$
$D \to bD \mid cD \mid \lambda$

3. For each of the following regular expressions, define the corresponding generative grammar (all over the alphabet Σ = {a, b, c, d}):

(i) {a i b j c jd i : i, j ≥ 1}.


3.(i) {a i b j c jd i : i, j ≥ 1}.

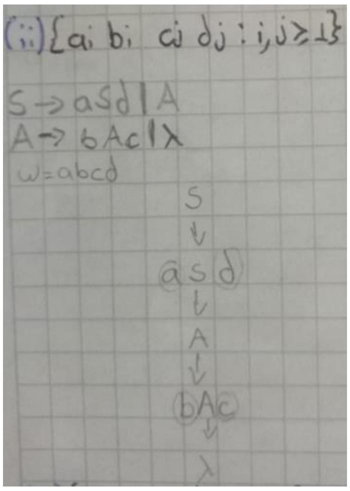
$S \to aSd \mid A$

$A \to bAc \mid bc$

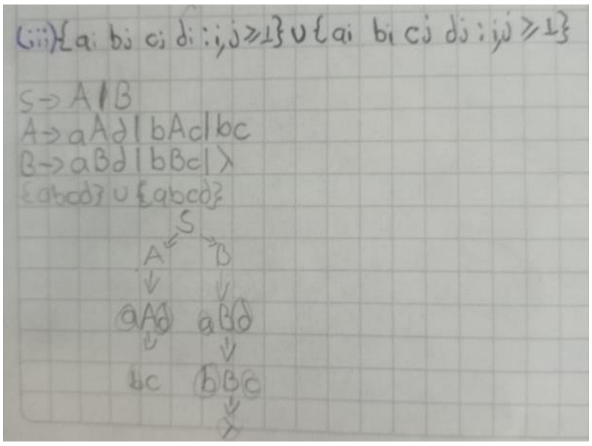
w=abcd

S
aSd
a A d
a bc d

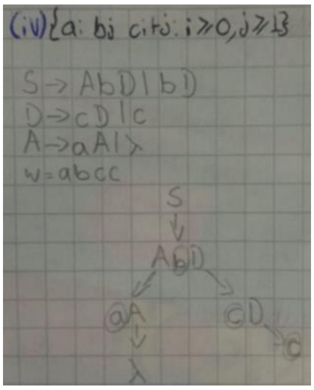
(ii) {a i b i c jd j : i, j ≥ 1}.



(iii) {a i b j c jd i : i, j ≥ 1} ∪ {a i b i c jd j : i, j ≥ 1}.



(iv) {a i b j c i+j : i ≥ 0, j ≥ 1}.

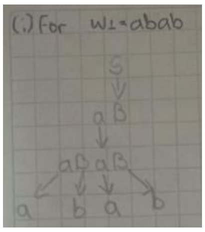


4. Be G a context-free grammar with the following productions:

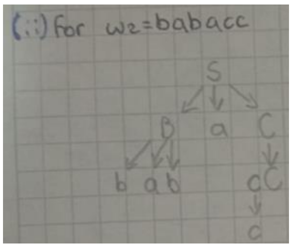
$$G = \begin{cases} S \to ABC \mid BaC \mid aB \\ A \to Aa \mid a \\ B \to BAB \mid bab \\ C \to cC \mid \lambda \end{cases}$$

Found derivation trees for the following strings:

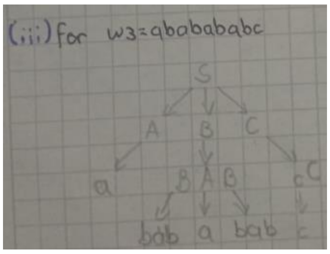
(i) w1 = abab.



(ii) w2 = babacc.



(iii) w3 = ababababc.

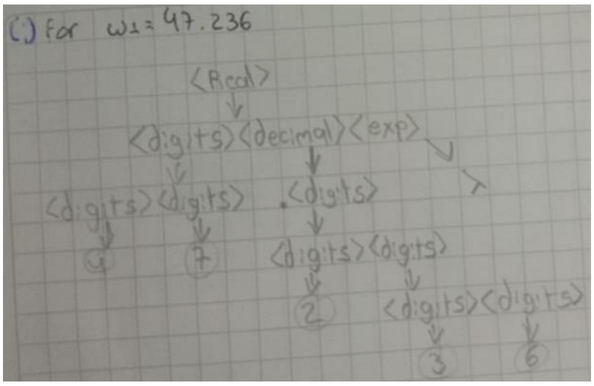


5. As follows there is a context-free grammar to generate real numbers without sign, the alphabet is Σ = {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, ., +, −, E}:

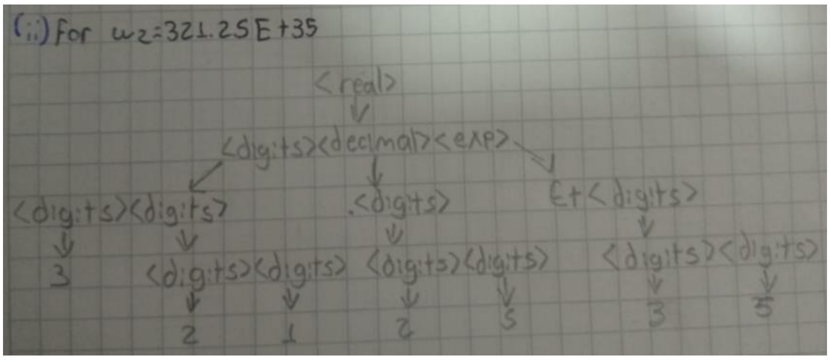
$$
\begin{array}{lcl}
< real > & \rightarrow & < digits >< decimal >< exp > \\
< digits > & \rightarrow & < digits >< digits >| \ 0 \ | \ 1 \ | \ 2 \ | \ 3 \ | \ 4 \ | \ 5 \ | \ 6 \ | \ 7 \ | \ 8 \ | \ 9 \\
< decimal > & \rightarrow & < digits >| \ \lambda \\
< exp > & \rightarrow & E < digits >| \ E+ < digits >| \ E- < digits >| \ \lambda
\end{array}
$$

Define the derivation tree for the following strings:

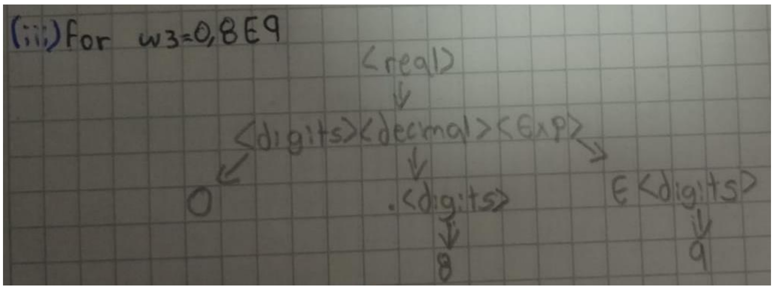
(i) w1 = 47.236



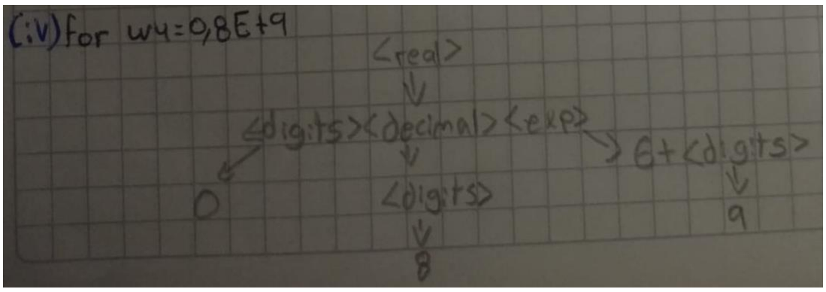
(ii) w2 = 321.25E + 35



(iii) w3 = 0.8E9



(iv) w4 = 0.8E + 9

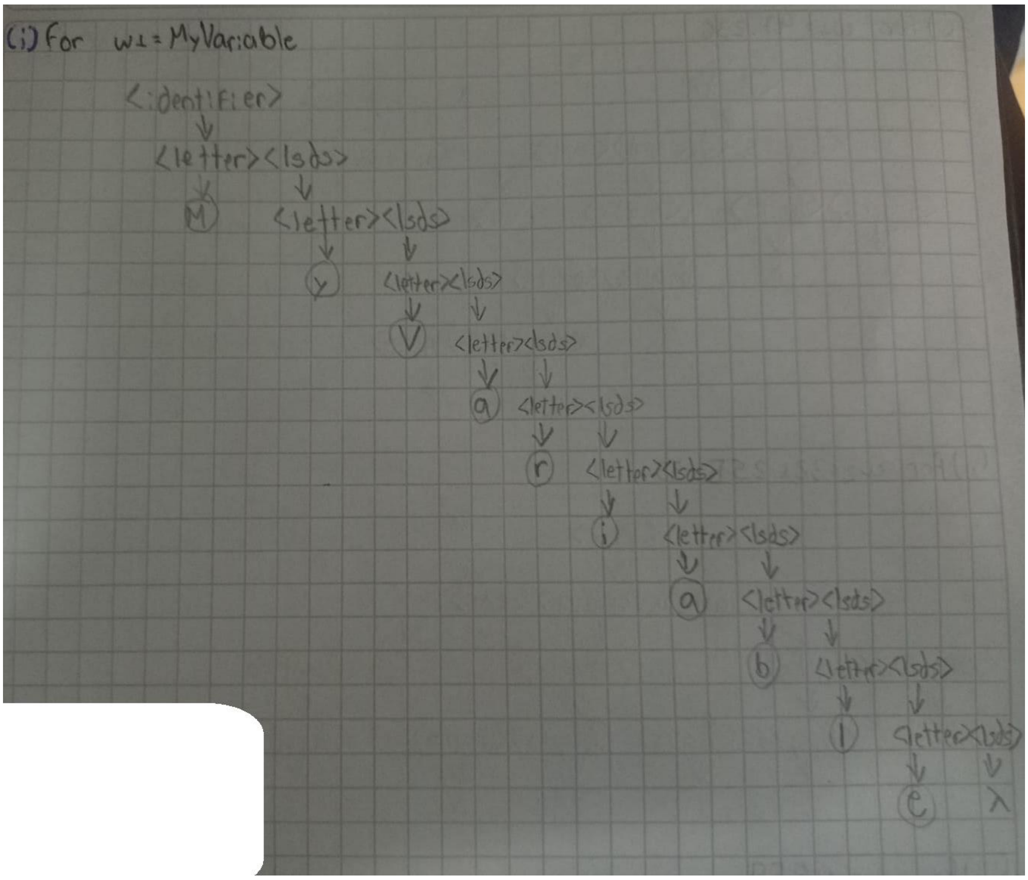


6. As follows there is a context-free grammar to generate identifiers, identifiers are strings of letters and digits, starting with a letter:

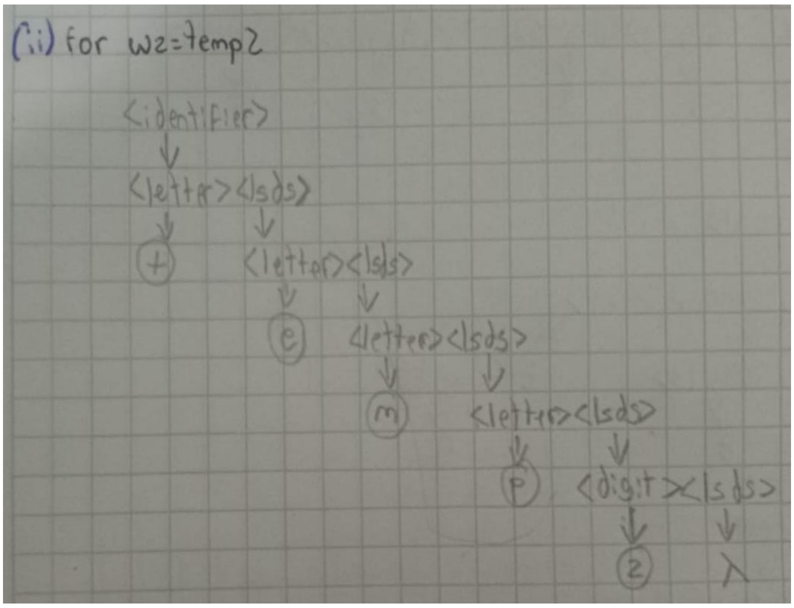
$$
\begin{array}{lcl}
<identifier> & \rightarrow & <letter><lsds> \\
<lsds> & \rightarrow & <letter><lsds> \mid <digit><lsds> \mid \lambda \\
<letter> & \rightarrow & a \mid b \mid c \mid \ldots \mid x \mid y \mid z \mid A \mid B \mid C \mid \ldots \mid X \mid Y \mid Z \\
<digit> & \rightarrow & 0 \mid 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9
\end{array}
$$

Define the derivation tree for the following names:

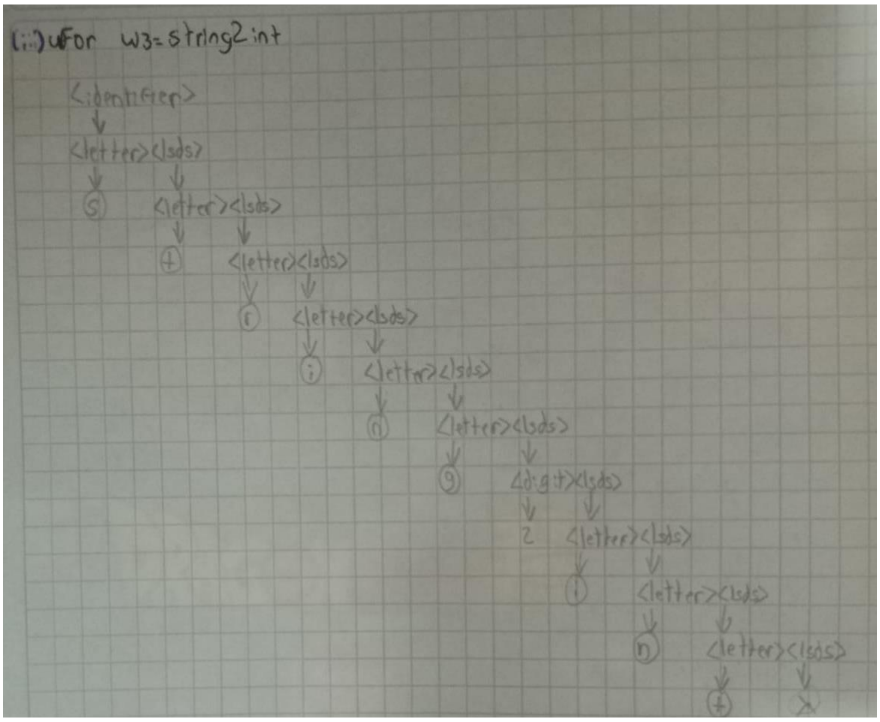
(i) w1 = MyVariable



(ii) w2 = temp2



(iii) w3 = string2int



(iv) w4 = 2NotAVariable

(:v) For w4: 2NotAVariable