Formal Languages and Automata Theory
Exercises Languages and Formal Grammars
Unit 4

Authors:

Araceli Sanchis de Miguel
Agapito Ledezma Espino
Jose A. Iglesias Martínez
Beatriz García Jiménez
Juan Manuel Alonso Weber

* Several exercises are based on the ones proposed in the following books:

- Enrique Alfonseca Cubero, Manuel Alfonseca Cubero, Roberto Moriyón Salomón. *Teoría de autómatas y lenguajes formales.* McGraw-Hill (2007).
- Manuel Alfonseca, Justo Sancho, Miguel Martínez Orga. *Teoría de lenguajes, gramáticas y autómatas.* Publicaciones R.A.E.C. (1997).
- Pedro Isasi, Paloma Martínez y Daniel Borrajo. *Lenguajes, Gramáticas y Autómatas. Un enfoque práctico.* Addison-Wesley (1997).

Universidad Carlos III de Madrid
www.uc3m.es

1. Create a grammar to generate the following languages:

a) **{ a, aa, aaa }**
b) **{ a, aa, aaa, aaaa, aaaaa, …)**
c) **{ λ, a, aa, aaa }**
d) **{ λ, a, aa, aaa, aaaa, aaaaa, …)**
The notation used to denote each one of the languages is:
a) **{ a^n | n ∈ [1, 3] }**
b) **{ a^n | n > 0 }**
c) **{ a^n | n ∈ [0, 3] }**
d) **{ a^n | n ≥ 0 }**

<u>**Solution**</u>:

a)
G=({a}, {S}, S, P} where:
P={S::=a | aa | aaa}

b)
G=({a}, {S, A}, S, P} where:
P={S::=A
A::= a | aA}

c)
G=({a}, {S}, S, P} where:
P={S::= λ | a | aa | aaa}

d)

G=({a}, {S, A}, S, P} where:
P={S::= λ |A
A:= a | aA}

2. Given the grammars G=({c,d}, {S,A,T}, S, P_i} where:

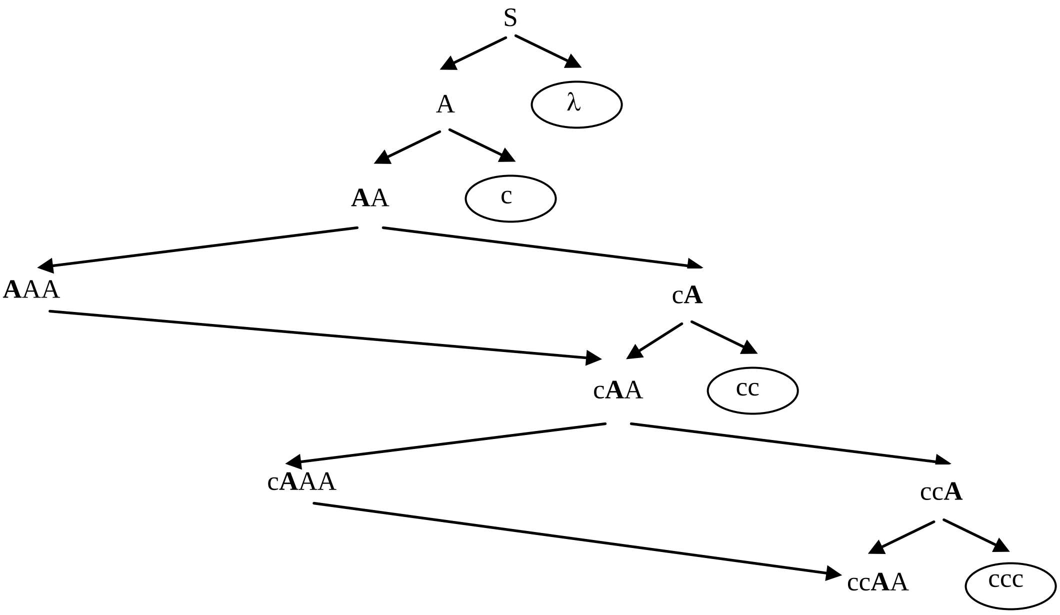
G_1	G_2	G_3	G_4	G_5
Σ_T = {c} Σ_{NT} = {S, A}	Σ_T = {c,d} Σ_{NT} = {S, A}	Σ_T = {c} Σ_{NT} = {S, A}	Σ_T = {c,d} Σ_{NT} = {S, A,T}	Σ_T = {c,d} Σ_{NT} = {S, A}
P_1: S→λ | A A→AA| c	P_2: S→λ | A A→cAd| cd	P_3: S→λ | A A→AcA| c	P_4: S→cA A→d | cA| Td T→Td | d	P_5: S→λ | A A→Ad| cA | c| d

Determine the associated language.

<u>**Solution**</u>:

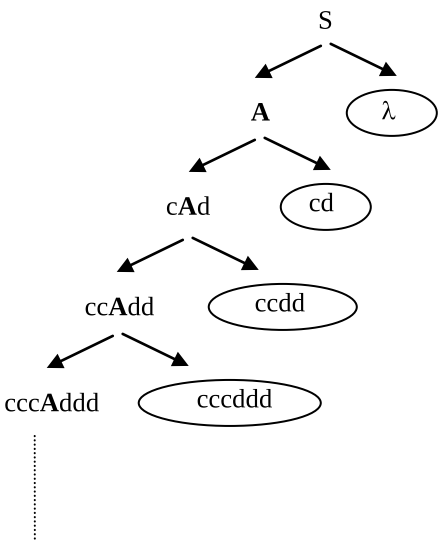
L(G) = {x / S*→ x AND x∈Σ* }

a) S::=λ | A
A::=AA | c



L(G_1)={ λ, c, cc, ccc, …. }= {λ, c^n} with n=1, 2, 3, ……

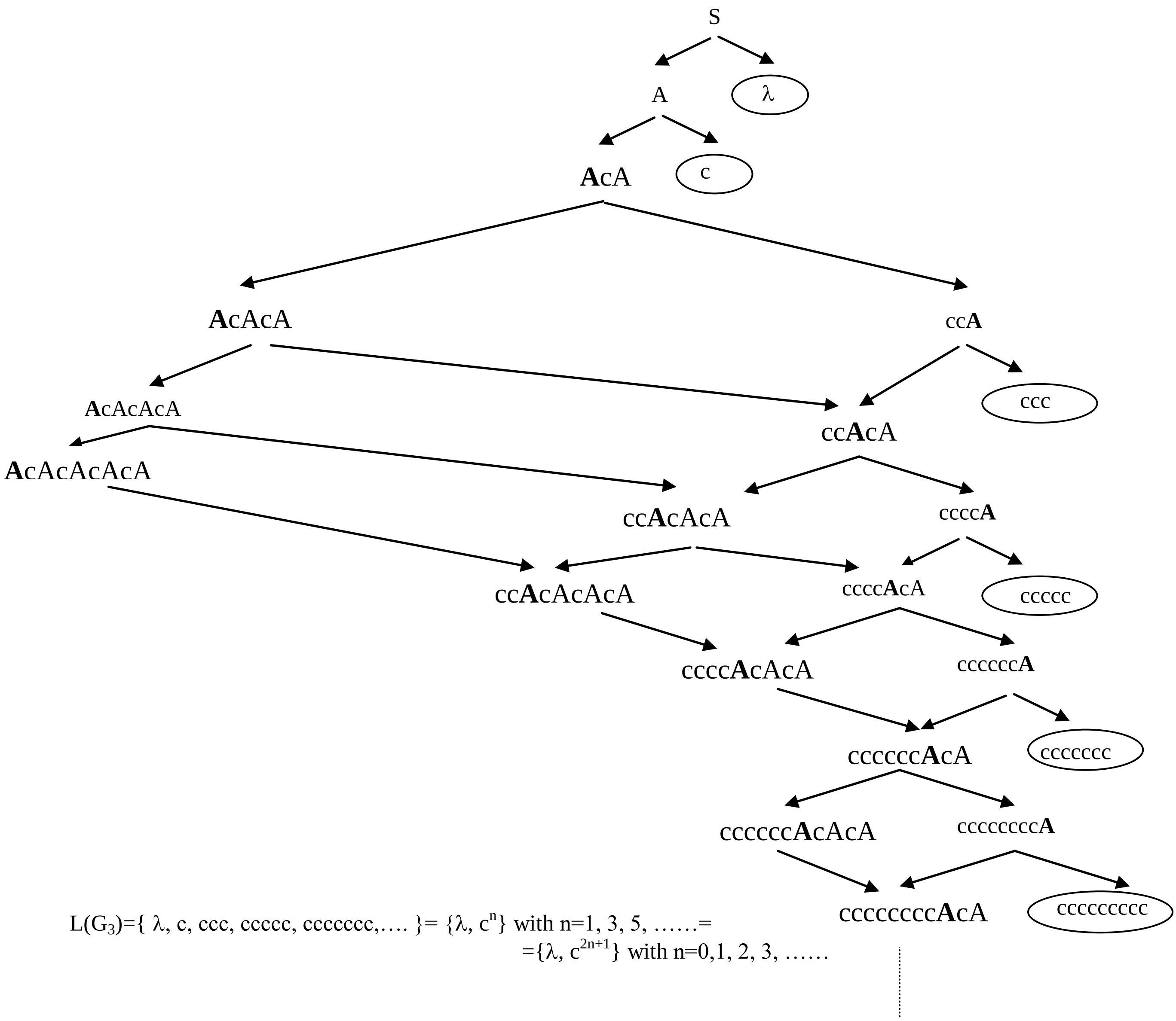
b) S::=λ | A
A::=cAd | cd



L(G_2)={ λ, cd, ccdd, cccddd, …. }= {λ, c^nd^n} with n=1, 2, 3, ……

Universidad
Carlos III de Madrid
www.uc3m.es

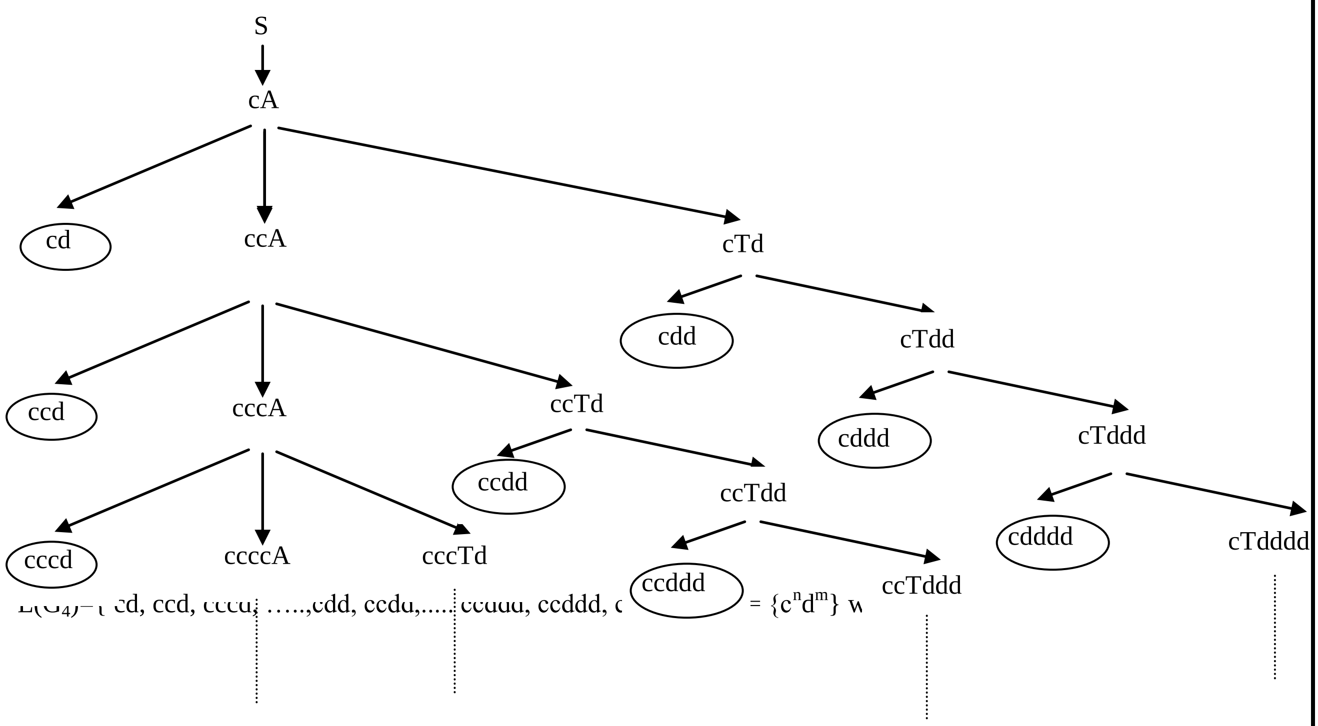
c) S::=λ | A
A::=AcA | c



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

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

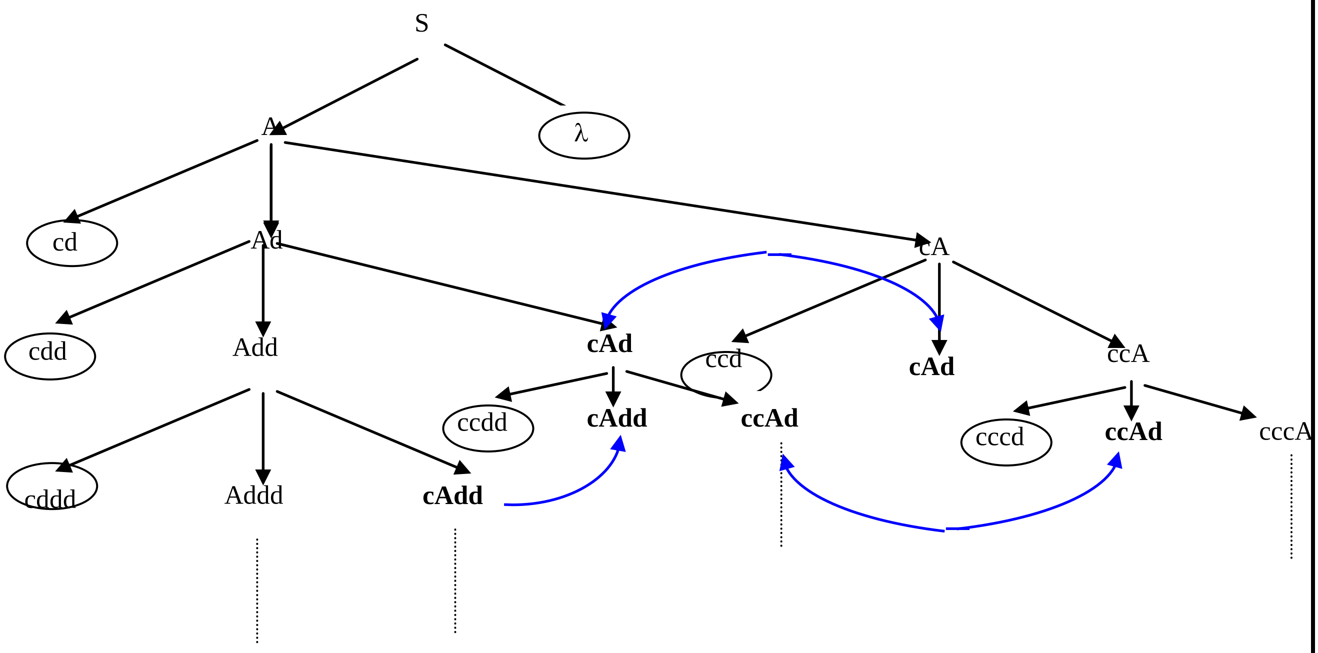
$L(G_d)=\{c^n d^m \ / \ \ n,m \geq 1\}$



L(G4)={ cd, ccd, cccd, …..,cdd, ccdd,..... ccddd, ccdddd, c … = $\{c^n d^m\}$ w

e) S::=λ | A
A::=cd |Ad | cA

L(G_e)= {λ, $c^n d^m$ / n,m≥1}



L(G_5)={ λ, cd, cdd, cddd,...... ccd, ccdd, ccddd,…. }= {λ, $c^n d^m$} with n,m>=1

3. Determine the type of the following grammars into the Chomsky Hierarchy. Justify your answer.

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=({ house, garden, cat}, {S, CASTLE, FOREST, TIGER }, S, P),
P={ S::=TIGER garden, S::=FOREST CASTLE, FOREST::=λ, garden CASTLE TIGHER house::=garden FOREST TIGER house, cat CASTLE FOREST::=cat FOREST house TIGER FOREST, FOREST::=TIGER house, FOREST::=garden }

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}

Solution:

a) Type-0.

b) Type-1.

c) Type-0.

d) Type-2.

e) Type-0.

4. Given the gramar G,

G=({a,b,c}, {S,A,B}, S, P), P={S::=λ, S::=aAc, A::=aA, A::=Ac, A::=B, B::=b, B::=Bb}

It is required:

a) Specify the type of G in the Chomsky Hierarchy. Justify your answer.

b) Determine the language L generated by the grammar.

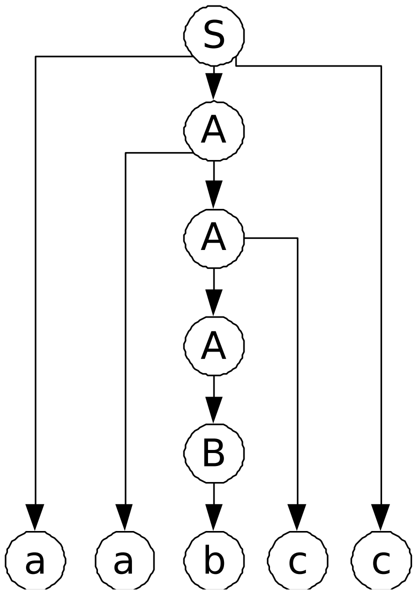
c) Construct two different derivation trees for a word in L(G).

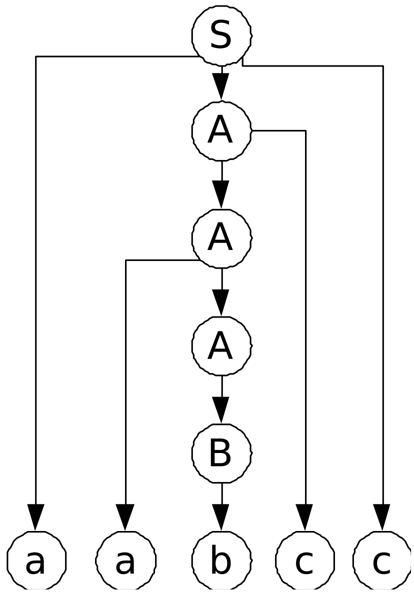
d) Verify if the following sentential forms are valid in G, and write a derivation chain to generate the valid ones.

d.1.- aaAcc
d.2.-ac
d.3.-ababBcc
d.4.-abbccc

Solution:

a) It id s Type-2 grammar.
b) L(G)={$a^p b^q c^r$, p=q=r=0 or p,q,r>0}.
c) word=aabcc





S → aAc → a**aAc** → aaAcc → aaBcc → aabcc
S → aAc → a**Acc** → aaAcc → aaBcc → aabcc

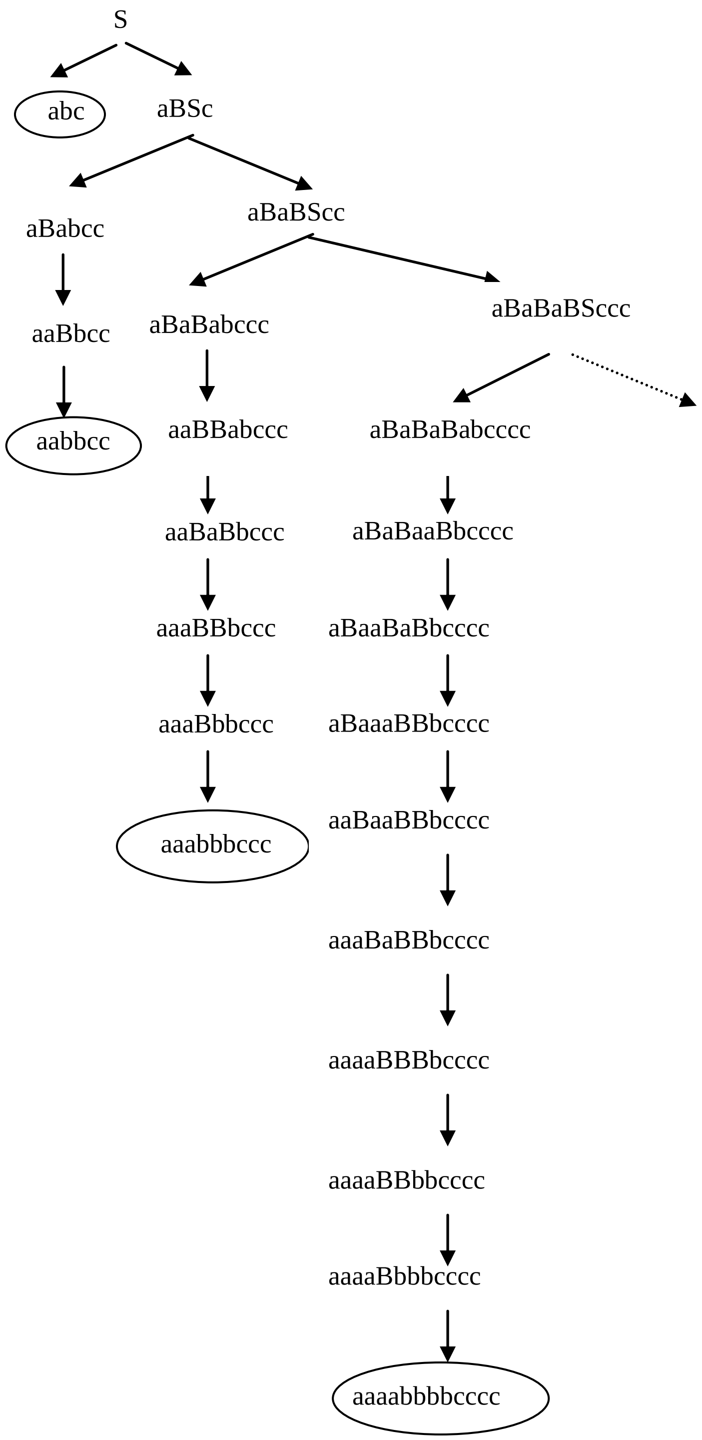
d)

d.1.- **aaAcc**: S → aAc → aaAc → aaAcc
d.2.- **ac:** not valid
d.3.- **ababBcc:** not valid
d.4.- **abbccc:** S → aAc → aAcc → aAccc → aBccc → aBbccc → abbccc

5. Obtain a type-0 grammar for the language $L=\{a^n b^n c^n \ / \ n \geq 1\}$.

<u>Solution</u>:

S::= abc| aBSc
Ba::=aB
Bb::=bb



6. Obtain an equivalent well formed grammar for the following one:

G = ({a,b,c,d}, {X,Y,Z,O,P,Q,A}, Z, P),
P = { Z::=Z, Q::=OP, X::=aa, Z::=aX, Y::=aa, Z::=Ya, O::=b, Z::=aaa,
P::=QO, Q::=d, P::=c, O::=PQ}

__Solution__:

G' = ({a}, {X,Y, Z}, Z, P'), P' = {Z::=aaa, Z::=aX, Z::=Ya, X::=aa, Y::=aa}

7. Given the following left-linear grammar G, obtain an equivalent right-linear G' grammar.

G=({0,1},{A,S},S,P)
P={ S ::= 1 | A1; A ::= S0}

__Solution__:

G LI = ({a,b}, {A,S,B}, S, P), P={S::=1|1B, B::=0A, A::=1|1B}

8. Given the gramar G:

G = ({e,f,g,z,a,b,d}, {Y, X, E, A, D, I, G}, A, P),
P = { A::=a
E::=b
A::=azb
A::=aX
E::=E
G::=g
X::=XE
D::=eI
X::=z
Y::=b
I::=fG
X::=Xb
E::=d }

a) Transform to CNF detailing the process followed.
b) Determine whether the words 'abz' y 'azdbb' are included in the language generated by G. If this is the case, generate a parse tree for the included words. If not, justify the not inclusion in the language.

__Solution__:

a)
G' '= ({a,b,d,z}, {A, B, C, D, E, F, X}, A, P"),
P" = {
A::=a
A::=BC
A::=BX
B::=a
C::=DF
D::=z
E::=b
E::=d
F::=b
X::=XE
X::=XF
X::=z
}

b)

A → **BX** → **a**X → a**XF** → a**XF**F → a**XE**FF → a**z**EFF → az**d**FF → azd**b**F → azdb**b**

9. Obtain a grammar in CNF equivalent to the following grammar:

G = ({a, b, c},{S, A, B, C, D, E), S, P)
P = { S::=AaB | Cbb | B
A::=Aa | cD
B::=a | Ba | λ
C::=Sa | a | abB
D::=aaA
E::=aa }

Solution:

G = ({a, b},{S, B,C,D,E,F,G), S, P)
P = { S ::= CD | a | BF | λ
B ::= a | BF
C ::= SF | a | FG | FE
D ::=EE
E ::= b
F ::= a
G ::= EB}

10. Given the gramar G calculate an equivalent grammar in GNF.

G = ({a,b}, {S}, S, P), donde P = {S::=aSb | SS | λ}

Solution:

G' = ({a,b}, {S,X,B}S, P'})
P' = {
S ::= aSB | aSbB | λ
X ::= aSBX | aSBXX | aSB
B ::= b
}