Homework #6 *due December 13, 2022 before recitation*

Question 1

Consider the CFG $\boldsymbol{G = (V, \Sigma, R, S)}$ *where* $\boldsymbol{V = \{S,A,B,C,D,E\}}$ *,* $\boldsymbol{\Sigma = \{a,b,c\}}$ *and* ***R*** *is as given below*

R*:*

$S \rightarrow AE \mid EB \mid C$

$A \rightarrow aA \mid a$

$B \rightarrow Bb \mid b$

$C \rightarrow Cc$

$D \rightarrow aCb \mid a \mid b \mid c$

$E \rightarrow aEb \mid e$

(a) *Remove all the null productions of* ***G****, if any, and call the result* $\boldsymbol{G_1}$**.**

(b) *Remove all the unitary productions of* $\boldsymbol{G_1}$*, if any , call the result* $\boldsymbol{G_2}$**.**

(c) *Remove all the non-generative and non-reachable symbols of this grammar, if any, and call the result* $\boldsymbol{G_3}$**.**

(d) *Compute the Chomsky Normal Form of* $\boldsymbol{G_3}$ *using your results above.*

(e) *State in the simplest possible way the language generated by* ***G***

Question 2

Consider the alphabet ***T*** *of terminals consisting of* ***3*** *pairs of matching left and right parentheses of three types****,*** *namely :* ***{ , } , [,] , (,)***

(a) *Describe a CFG ,* $\boldsymbol{G = (V, T, R, S)}$ *such that* $\boldsymbol{L_G}$ *has the following properties:*

(i) every left parenthesis is balanced by a <u>distinct</u> right parenthesis somewhere on its right side and of its own type; (ii) a priority rule holds : no curly parenthesis - i.e. ***{*** *or* ***}*** *- is contained within a rectangular pair – i.e.* ***[ ]*** *- or a plain pair – i.e.* ***( )*** *- ; and no rectangular parenthesis is contained within a plain pair; (iii) empty string is not a member of* $\boldsymbol{L_G}$*.*

(b) *Using your grammar find a parse tree that derives the string:* ***{ ( ) [( )] } { }***

Main Text: Exercise 6.4.1 (a),(c) ; 6.4.2

Question 2

$G(N, T, P, S)$

$S \rightarrow SS \mid \{A\} \mid [B] \mid (C)$

$A \rightarrow AA \mid \{A\} \mid [B] \mid (C) \mid \epsilon$

$B \rightarrow BB \mid [B] \mid (C) \mid \epsilon$

$C \rightarrow CC \mid (C) \mid \epsilon$

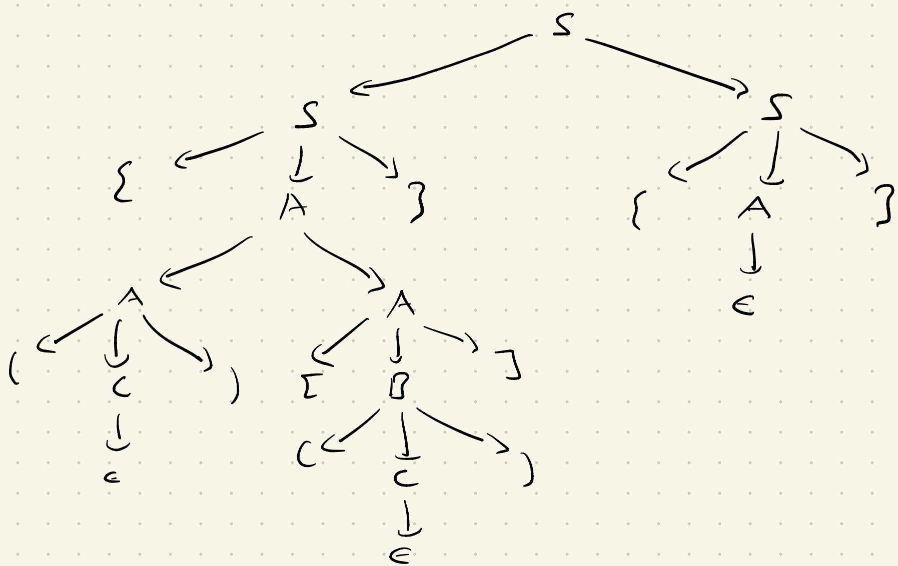
N - Non terminal Symbol - S, A, B, C

T - Terminal Symbol - {, }, [,], (,)

S - Start Symbol

P - Productions -

The Parse tree of {()[()]}{} is given below



Exercise 6.4.1.

a)

1. $\delta(q, 0, z_0) = \{(q, xz_0)\}$
2. $\delta(q, 0, x) = \{(q, xx)\}$
3. $\delta(q, 1, x) = \{(q, x)\}$
4. $\delta(q, \lambda, x) = \{(p, \lambda)\}$
5. $\delta(p, \lambda, x) = \{(p, \lambda)\}$
6. $\delta(p, 1, x) = \{(p, xx)\}$
7. $\delta(p, 1, z_0) = \{(p, \lambda)\}$

Rule 3, Rule 4 violating ② of DPDA.

Rule 5, Rule 6 " " " ".

Rule 2, Rule 4 " " " ".

Hence, given PDA is not deterministic.

Conditions of DPDA

① $\delta(q, a, X)$ has at most one member for any $q \in Q$, $a \in \Sigma$ or $a = \lambda$ and $X \in \Gamma$

② If $\delta(q, a, x)$ is nonempty for some $a \in \Sigma$. Then, $\delta(q, \lambda, x)$ must be empty.

→ The first condition requires that for any input symbol, any stack top at most one move can be made.

→ The second condition requires that when λ move is possible for some configuration no input consuming alternative is available.

c) 1. $\delta(q, 1, z_0) = \{(q, xz_0)\}$ → Rule 2 and Rule 4 are violating the ② of DPDA

2. $\delta(q, 1, x) = \{(q, xx)\}$ → Rule 3 " " " " " " " "

3. $\delta(q, 0, x) = \{(p, x)\}$

4. $\delta(q, \lambda, x) = \{(q, \lambda)\}$

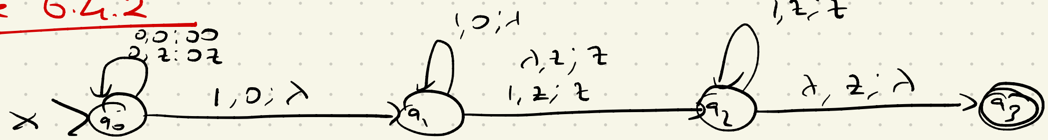
5. $\delta(p, 1, x) = \{(p, \lambda)\}$

6. $\delta(p, 0, z_0) = \{(p, z_0)\}$

Hence, given PDA is not DPDA!

Exercise 6.4.2

a)



b)