Assignment -02 September, 2020

(a) Design an equivalent left-recursion free and left factored grammar of the following grammar G.

$P \rightarrow (E) \mid a$

$E \rightarrow E, P \mid P$

ns: Applying left recursion removal,

$P \rightarrow (E) \mid a$

$E \rightarrow PE'$

$E' \rightarrow , PE' \mid \varepsilon$

(b) Construct FIRST and FOLLOW sets for non-terminals of the resulting grammar.

ns:

FIRST(P) = {(, a}

FIRST(E) = {(, a}

FIRST(E') = {, , ε}

FOLLOW(P) = {$, , ,)}

FOLLOW(E) = {)}

FOLLOW(E') = {)}

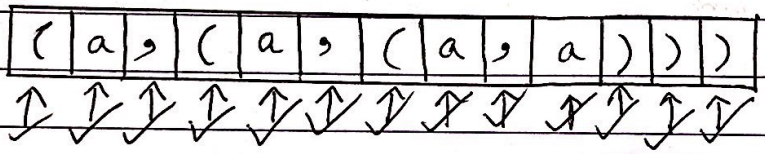
(c) Construct LL(1) Parse Table for the resulting grammar.

Ans:

	$	,	(	)	a
P			P → (E)		P → a
E			E → PE'		E → PE'
E'		E' → , PE'		E' → ε	

(d) Validate the following Syntax through Predictive Parsing using the constructed Parse Table.

(a, (a, (a, a)))



Output	Stack		
	~~(~~	~~P~~	
P → (E)	~~E~~	~~E'~~	
E → PE'	~~)~~	~~a~~	
P → a	~~)~~	~~P~~	
E' → , PE'	~~P~~	~~E'~~	
P → (E)	~~E'~~	~~)~~	
E → PE'	~~a~~	~~E~~	
P → a	~~P~~	~~)~~	
E' → , PE'	~~E'~~	~~)~~	
P → (E)	~~(~~	~~P~~	
E → PE'	~~E~~	~~E'~~	
P → a	~~)~~	~~a~~	
E' → , PE'	~~P~~	~~P~~	~~a~~
P → a	$	~~E'~~	~~)~~

E' → ε
E' → ε
E' → ε