Asignatura	**Sintaxis y Semántica de Lenguajes**
Carrera	**Ingeniería en Sistemas de Información**
Alumno	

10/12/2020

Evaluación Final

P1	Construya la TAS correspondiente a la siguiente CFG [60 ptos]

W → WF | KM | CM | MW@W | MDM | W*H
H → C | D | [W]
K → [W] | C
P → P | Pa | E | (P)
F → FP | aL | ε
E → EbG | GaW | aPb
D → #D | D/ | #
M → ε | MG
C → Cg | Mg | aB | uC

P2 (40 ptos)	En base a la CFG de **P1**: [15 ptos] A. Construya una gramática equivalente en CNF [20 ptos] B. Construya una gramática equivalente en GNF [5 ptos] C. Construya un PDA equivalente

Limpiamos la gramática

W → [W] | C | W@W | D | W*H
H → C | D | [W]
D → #D | D/ | # #####D D/////
C → Cg |g | uC

Sacamos ambigüedad de W, D, C

W → W@H | W*H | H
H → C | D | [W]
D → NB
N → #N | #
B → /B | eps
C → UG
U → uU | eps
G → gG | g

Sacamos RI de W
fórmula de RI
A → A alfa | beta

A → beta W
W → alfa W | epsilon

Gramática en LL1 sacar ambigüedad, RI y debe estar factorizada

W → H Z
Z → @HZ | *HZ | eps
H → C | D | [W]
D → NB
N → #M
M → N | eps
B → /B | eps
C → UG
U → uU | eps
G → gL
L → G | eps

Para W → H Z
prim(HZ) = {u,g, #, []}
prim (H) = {u,g, #, []}
prim (C) = {u,g}
prim (D)= {#}
prim ([W]) = {[}

Para Z → @HZ
prim (@HZ) ={@}

Para Z → *HZ
prim (*HZ) ={*}

Para Z → eps
prim (eps) = {eps}
sig (Z) ={$,]}
sig(W) {$,]}

Para H → C
prim (C) ={u,g}

Para H → D
prim (D) = {#}

Para H → [W]
prim ([W]) = {[}

Para D → NB
prim (NB) ={#}

Para N → #M
prim (#M) = {#}

Para M → N
prim (N) = {#}

Para M → eps
prim (eps) = {eps}
sig (M) = {$,], @,*, /}
sig (N) = {$,], @,*, /}
prim (B) = {/,eps}
sig (B) = {$,], @,*}
sig (D) = {$,], @,*}
sig (H)= {$,], @,*}
prim (Z) = {@,*,eps}
sig (Z) ={$,]}
sig (W) ={$,]}

Para B → /B
prim (/B) ={/}

Para B → eps
prim (eps) ={eps}
sig (B) = {$,], @,*}

Para C → UG
prim (UG) = {u,g}

Para U → uU
prim (uU) ={u}

Para U → eps
prim (eps) ={eps}
sig (U)= {g}

Para G → gL
prim (gL) ={g}

Para L → G
prim (G) = {g}

Para L → eps
prim (eps) = {eps}
sig (L)= {$,], @,*}
sig (G) = {$,], @,*}
sig (C)= {$,], @,*}
sig (H)= {$,], @,*}

	@	*	[	]	#	/	u	g	$
W			HZ		HZ		HZ	HZ	
Z	@HZ	*HZ		eps					eps
H			[W]		D		C	C	
D					NB				
N					#M				
M	eps	eps		eps	N	eps			eps
B	eps	eps		eps		/B			eps
C							UG	UG	
U							uU	eps	
G								gL	
L	eps	eps		eps				G	eps

CNF, GNF y PDA

Limpiamos la gramática

W → [W] | C | W@W | D | W*H
H → C | D | [W]
D → #D | D/ | #
C → Cg |g | uC

CNF (2 variables o 1 terminal)

W → XZ | WA | WE |ND | DV | # |CG | g | UC
X → [
Z → WY
Y →]
A → BW
B → @
E → FH
F → *
H → XZ |ND | DV | # | CG | g | UC
D → ND | DV | #
N→ #
V → /
C → CG | g | UC
G → g
U → u

GNF (1 terminal seguido de cero o más variables)

W → [W] | C | W@W | D | W*H
H → C | D | [W]
D → #D | D/ | #
C → Cg |g | uC

Eliminamos RI de W y epsilon de la nueva variable //cálculo auxiliar

W → [W] | C | W@W | D | W*H

W → [W] Z | CZ |DZ | [W] | C|D
Z → @WZ | *HZ |@W | *H

Eliminamos RI de D //cálculo auxiliar

D → #D | D/ | #

D → #DX | #X | #D |#
X → /X | /

Eliminamos RI de C //cálculo auxiliar

C → Cg |g | uC

C → gU | uCU | g | uC
U → gU | g

Armo la gramática original a partir de la gramática limpia y de los cálculos auxiliares

W → [W PZ | gUZ | uCUZ | gZ | uCZ |#DXZ | #XZ | #DZ |# Z| [W P | gU | uCU | g | uC| #DX | #X | #D |#
Z → @WZ | *HZ |@W | *H
H → gU | uCU | g | uC | #DX | #X | #D |# | [W P
D → #DX | #X | #D |#
X → /X | /
C → gU | uCU | g | uC
U → gU | g
P →]

PDA

Q	SIGMA U EPS	GAMMA	2**QXGAMMA*
0	[	W	{(0,WPZ), (0,WP)}
0	g	W	{(0,UZ),(0,Z), (0,U), (0,EPS) }
0	u	W	{(0,CUZ),(0,CZ), (0,CU), (0,C) }
0	#	W	{(0,DXZ),(0,XZ), (0,DZ), (0,Z) ,(0,DX),(0,X), (0,D), (0,EPS)}
0	@	Z	{(0,WZ),(0,W)}
0	*	Z	{(0,HZ),(0,H)}
0	g	H	{(0,U),(0,EPS)}
0	u	H	{(0,CU),(0,C)}
0	#	H	{(0,DX),(0,X), (0,D), (0,EPS) }
0	[	H	{(0,WP)}
0	#	D	{(0,DX),(0,X), (0,D), (0,EPS) }
0	/	X	{(0,X),(0,EPS)}
0	g	C	{(0,U),(0,EPS)}
0	u	C	{(0,CU),(0,C)}
0	g	U	{(0,U),(0,EPS)}
0	]	P	{(0,EPS)}