



Primer Semestre 2020

Sección	Catedrático	Tutor académico
A-	Inga. Damaris Campos de López	Fernando Mérida
B-	Inga. Zulma Karina Aguirre Ordoñez	Luis Javier Yela Quijada
A+	Ing. Otto Amílcar Rodríguez	Elmer Real
B+	Ing. David Estuardo Morales	José Veliz

Enunciado de Proyecto 2

Objetivos:

- Que el estudiante implemente una solución de software con base en los distintos paradigmas de programación vistos en clase y laboratorio.
- Conocer y aprender sobre las gramáticas independientes del contexto, logrando implementar un autómata de pila sencillo, en el lenguaje Python.

Descripción:

Dado a la capacidad de los autómatas de pila para aceptar lenguajes que no pueden aceptar los autómatas finitos por ejemplo el lenguaje $L\{x^n y^n : n \in \mathbb{R}\}$, se le pide a usted crear o modificar su solución del primer proyecto, haciéndola más potente y así pueda reconocer cosas que antes le eran imposibles, pero que ahora utilizando un autómata de pila a partir de una gramática tipo 2 ingresada a través de consola, podrá hacerlo sin problemas, más adelante se detallara la sintaxis, al igual que en el proyecto anterior podrá ser capaz de validar cadenas pero dado que ahora lo hará con una gramática tipo 2 tendrá un autómata mucho más potente y podrá reconocer cosas que con un AFD o una gramática regular le hubieran resultado imposibles.

Para la construcción de un autómata de pila para una gramática tipo 2, se debe de realizar según lo que se vea en clase y también puede consultar el libro de “teoría de la computación” de Glenn Brooksbeer, en donde se explica esto en el capítulo 2.

Un árbol de análisis sintáctico no es más que un árbol cuyos nodos representan terminales y no terminales de la gramática, los hijos de cada nodo no terminal son los símbolos que reemplazan a ese no terminal en la derivación (ningún símbolo terminal puede ser un nodo interior del árbol ni ningún símbolo no terminal puede ser una hoja). Más adelante se mostrará un árbol de sintáctico.

Características del programa

- La pantalla inicial:

Deberá de ser una carátula en donde se indique el nombre del curso, la sección, y su carné, al presionar la tecla enter deberá de dirigirse a otra pantalla en la cual habrá un menú con las opciones:

- Agregar un submenú para Gramáticas tipo2 y AP, adicional a las opciones que ya realiza el proyecto1:
 - Ingresar / modificar gramática: Limpia la pantalla y lo dirige al menú de gramática.
 - Generar Automata de Pila: Su única función es la de generar el autómata de pila correspondiente a la gramática y generar una imagen de éste.
 - Visualizar autómata: mostrará la imagen generada en el punto anterior y en consola desplegará la séxtupla de la cual está compuesto este autómata, la cual será detallada más adelante.
 - Validar cadena: le dirige a otra pantalla con las opciones para una cadena.
 - Regresar a la carátula: Regresa a la pantalla anterior.
- Salir: Detendrá la aplicación

- Menú Ingresar / modificar gramática:

Le pedirá el nombre de la gramática si este nombre ya existe se modificará la gramática si el nombre no existiera crea una nueva.

- Ingresar terminales: Estos pueden ser cualquier carácter alfanumérico o cualquier símbolo y se deben de ingresar uno por uno, no pueden ser letras mayúsculas.
- Ingresar no terminales: Estos serán ID que empiezan por una letra mayúscula.
- Ingresar producciones: Más adelante se detallará la sintaxis a usar.
- Borrar producciones: Le permitirá ingresar una producción para poder borrarla. Más adelante se detalla la manera de hacerlo
- No terminal inicial: se guarda el último no terminal inicial ingresado.
- Regresar al menú principal

- Validar cadena:

Pide un nombre para saber que gramática será la que va a utilizar.

1. Ingresar cadena
2. Resultado: En caso la cadena haya sido valida muestra la imagen del árbol sintáctico, caso contrario muestra un mensaje de error.
3. Reporte: este se explicará más adelante.

- Sintaxis para el ingreso de las producciones:

Cabe mencionar que no deberá de crear ningún analizador para leer las producciones pues estas siempre vendrán en la forma especificada, y deberá de usar el paradigma basado en objetos para guardar la información correspondiente a la gramática y autómata de pila.

$A > \alpha b \beta$

Donde:

- α y β son un conjunto de terminales y no terminales
 - b es un terminal.
 - Se seguirá usando la palabra épsilon para denotar el vacío.
- Borrado de producciones:
 - La notación de la producción que se va a eliminar es $A > \alpha \beta$,
Donde:
 - A es un no terminal
 - α, β son un conjunto de terminales y no terminales.
 - Ejemplo
Teniendo una gramática como la siguiente si yo quiero eliminar su segunda producción se debe hacer así:
 $A > a B C$
 $A > b X$
 Para borrar solo una producción yo debería de ingresar: $A > b X$
 Solo se puede borrar una producción a la vez.

Ejemplo de Gramática tipo 2:

$S > z M N z$

$M > a M a$

$M > z$

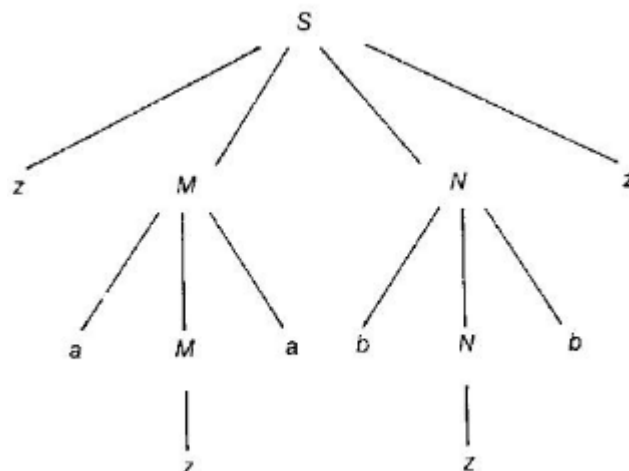
$N > b N b$

$N > z$

Ejemplo de entrada y de su árbol generado:

Entrada: $z a z a b z b z$

Árbol generado:



Autómata de pila:

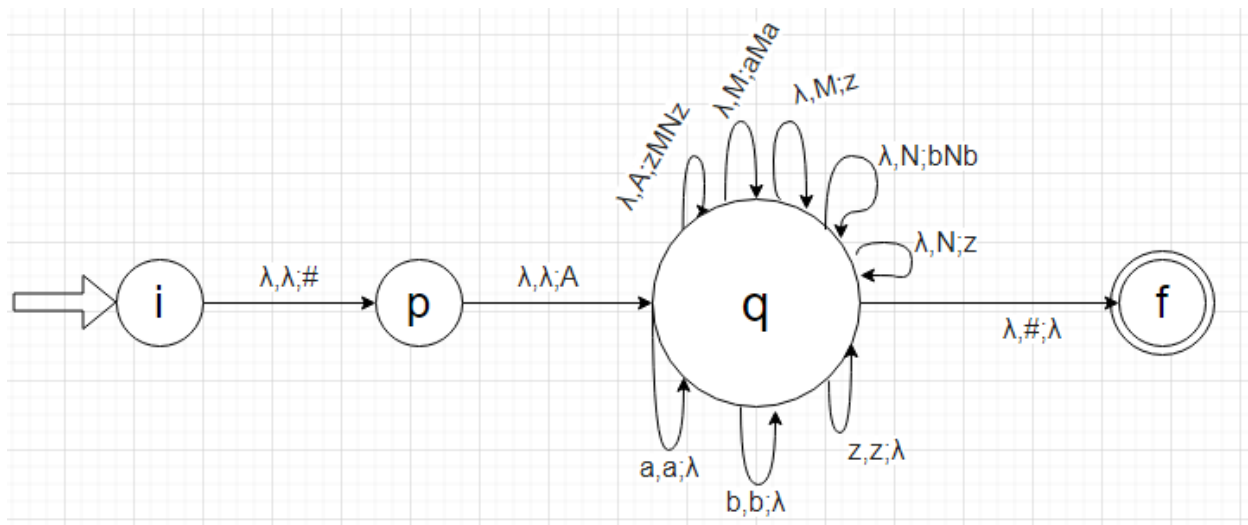
Un autómata de pila es una séxtupla de la forma: $(S, \Sigma, \Gamma, T, L, F)$

Donde:

- S : un conjunto finito de estados
- Σ : el alfabeto de la maquina
- Γ : un conjunto finito de símbolos de pila
- T : un conjunto finito de transiciones
- L : un estado inicial que pertenece al conjunto S
- F : un conjunto de estados de aceptación que es un subconjunto de S .

Para cada gramática G , independiente del contexto existe un autómata de pila M tal que $L(G)=L(M)$.

Ejemplo del autómata de pila generado a partir de la gramática anterior.



- $S: \{i, p, q, f\}$
- $\Sigma: \{a, b, z\}$
- $\Gamma: \{a, b, z, S, M, N, \#\}$
- $L: i$
- $F: f$
- $T:$
 - $i, \lambda, \lambda; p, \#$
 - $p, \lambda, \lambda; q, A$
 - $q, \lambda, A; q, zMNz$
 - $q, \lambda, M; q, aMa$
 - $q, \lambda, M; q, z$
 - $q, \lambda, N; q, bNb$
 - $q, \lambda, N; q, z$
 - $q, a, a; q, \lambda$
 - $q, b, b; q, \lambda$
 - $q, z, z; q, \lambda$
 - $q, \lambda, \#; f, \lambda$

Como lo que se le solicita realizar es un autómata de pila deberá de realizarlo utilizando su propia pila y realizando las instrucciones push o pop correspondientes, todas estas operaciones deberán ser hechas con el paradigma funcional dejando bien documentado todas sus funciones.

Reporte:

Este se generará con base en la entrada ingresada la cual debe de contener todos los movimientos en la pila que se llevó a cabo para la lectura de la cadena mostrando como se fue leyendo la cadena y las transiciones que se fueron haciendo en el proceso.

Ejemplo de movimientos en la pila y en la entrada con las transiciones correspondientes.

Validar la cadena con la el Autómata de pila generado para la gramática del ejemplo anterior.

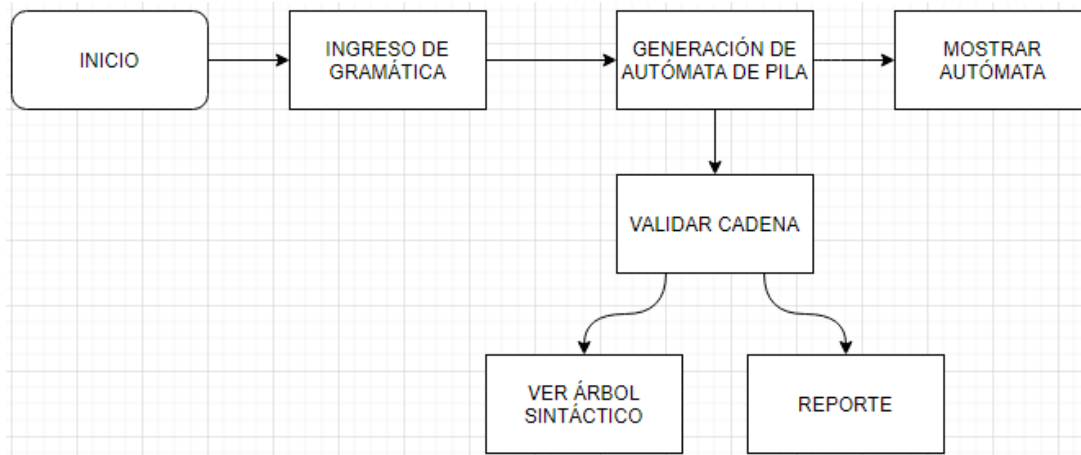
Entrada: z a z a b z b z

PILA	ENTRADA	TRANSICION
λ	z a z a b z b z	(i, λ , λ ; p, #)
#	z a z a b z b z	(p, λ , λ ; q, A)
A#	z a z a b z b z	(q, λ , A; q, zMNz)
zMNz#	z a z a b z b z	(q, z, z; q, λ)
MNz#	a z a b z b z	(q, λ , M; q, aMa)
aMaNz#	a z a b z b z	(q, a, a; q, λ)
MaNz#	z a b z b z	(q, λ , M; q, z)
zaNz#	z a b z b z	(q, z, z; q, λ)
aNz#	a b z b z	(q, a, a; q, λ)
Nz#	b z b z	(q, λ , N; q, bNb)
bNbz#	b z b z	(q, b, b; q, λ)
Nbz#	z b z	(q, λ , N; q, z)
zbz#	z b z	(q, z, z; q, λ)
bz#	b z	(q, b, b; q, λ)
z#	z	(q, z, z; q, λ)
#	-----	(q, λ , #; f, λ)
-----	-----	ACEPTACION

- NOTA: Este reporte debe de ser un archivo CSV, el separador que se usará es \$, quedando de la siguiente manera:
 - PILA\$ENTRADA\$TRANSICIÓN

(Esto con el fin de que el estudiante lo pueda mostrar en formato de tabla)

Flujo del programa



Entregables:

- Manual de Usuario
- Manual Técnico, debe explicar con un modelo la lógica de su programa.
- Código Fuente

Documentación a entregar de forma física el día de la calificación:

- Hoja de calificación (Original y una copia)

Notas importantes:

- El proyecto se debe desarrollar de forma individual.
- El proyecto deberá ser ejecutado utilizando la terminal de su sistema operativo.
- Este proyecto se deberá desarrollar utilizando Python.
- La entrega se realizará en Classroom
- Se debe de hacer uso de los paradigmas de programación vistos en clase y laboratorio.
- No se puede agregar o quitar algún símbolo en la sintaxis para el ingreso de las gramáticas.
- La calificación del proyecto será personal y durará como máximo 30 minutos, en un horario que posteriormente será establecido. Se debe tomar en cuenta que durante la calificación no podrán estar terceras personas alrededor, de lo contrario no se calificará el proyecto.
- No se dará prórroga para la entrega del Proyecto.
- Copia parcial o total del Proyecto tendrá una nota de 0 puntos, y se notificará a la escuela de sistemas para que se apliquen las sanciones correspondientes.
- En el caso de no cumplir con alguna de las indicaciones antes mencionadas, NO se calificará el proyecto; por lo cual, se tendrá una nota de cero puntos.

Fecha de entrega: 30 de abril de 2020 antes de las 23:59.