



PYTHON: GRAMMATICAS LIBRE DE CONTEXTO Y AUTOMATAS DE PILA MANUAL TECNICO

PROYECTO 2



ALBERTO GABRIEL REYES NING

LFP-LAB, SECCION A+
30/04/21

Contenido

INTRODUCCION.....	2
REQUISITOS DEL SISTEMA.....	2
AUTOMATA DE PILA.....	3
MODELO DE FLUJO DE DATOS	4
LOGICA	5

INTRODUCCION

Un programa diseñado para consola en Python. El programa carga un archivo '.txt' y almacena la información del archivo txt. La información contiene comandos que deben ejecutarse en las distintas listas, como ordenar la lista o buscar las posiciones de un determinado número dentro de su lista respectiva

REQUISITOS DEL SISTEMA

SO: Windows_NT x64 10.0.19041

Lenguaje de programación: Python versión 3.9.1 64-bit

Procesador de la computadora: Intel® Core™ i7-6700-HQ CPU @ 2.60GHz (8 CPUs), ~2.6GHz

Memoria física instalada (RAM): 16GB

IDE: Visual Studio Code 1.53.2

Extensiones para VSCode: Python Docstring Generator v0.5.4

Pylance v2021.2.3 L

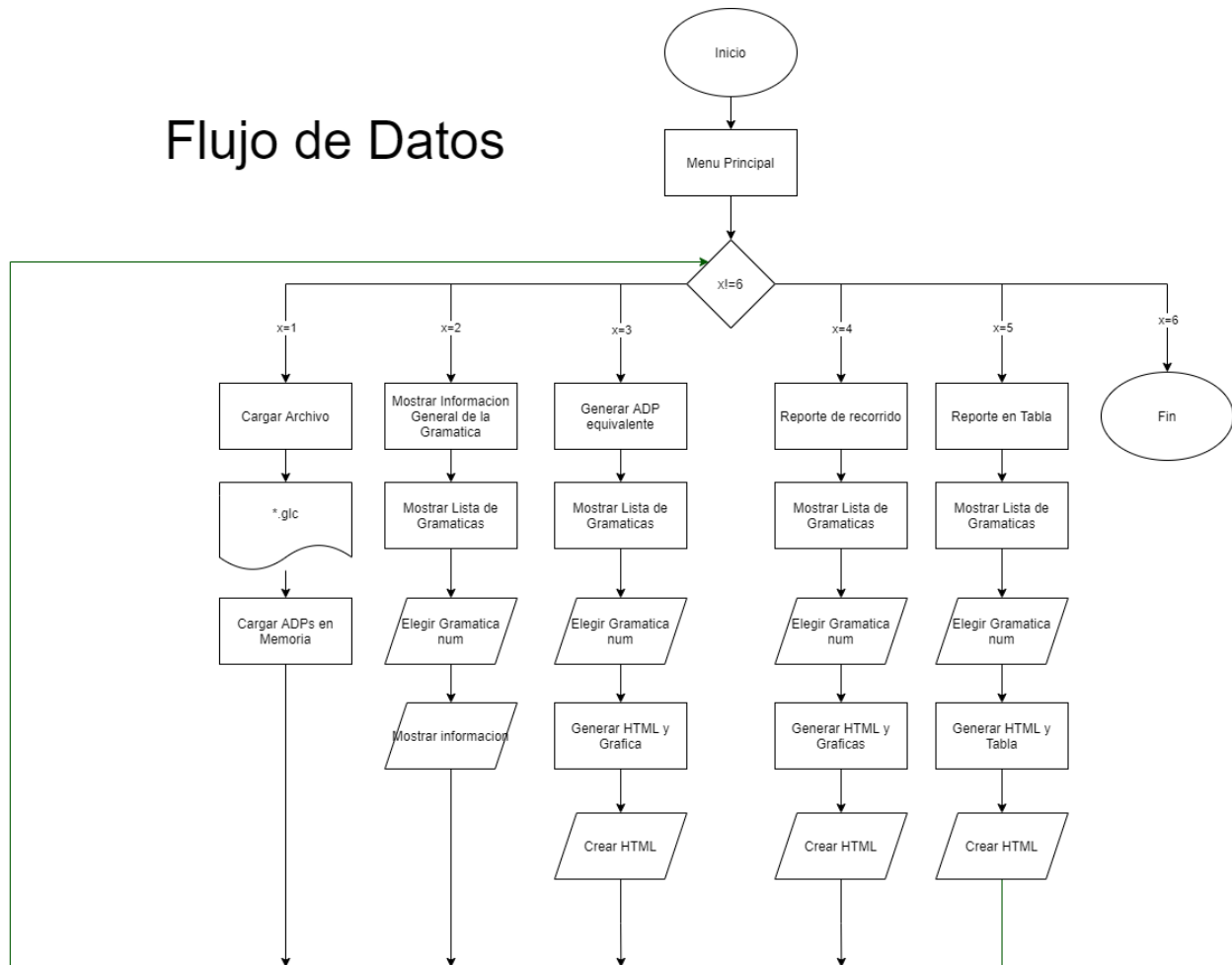
librerías: Tkinter

AUTOMATA DE PILA

Un autómata de pila es un autómata finito con memoria adicional llamado pila que ayuda a los autómatas de pila a reconocer lenguajes libres de contexto. En el programa, el autómata de pila utiliza listas para guardar los alfabetos del gramático. El autómata de pila tiene cuatro estados, cuales son 'i', 'p', 'q' y 'f'. Los estados 'i', 'p' y 'f' solo se utilizan para una función en el autómata. El estado 'q' contiene cuatro estados que varia basado en la pila y carácter de la cadena en esa transición. El primer estado verifique que el carácter en la pila es el termino inicial y utiliza la primera producción en la gramática. El segundo estado confirme si la pila llevo al termino final '#' y que se recorrió toda la cadena. El tercer estado confirme que el carácter en la pila es un carácter terminal que confirme con el carácter de la cadena en esa transición. Si confirme, el carácter de la pila se sale y se avanzó al siguiente carácter en la cadena. El cuarto estado confirme que el carácter en la pila es un carácter non-terminal y utilizando la gramática, ejecuta la producción apropiada.

MODELO DE FLUJO DE DATOS

Flujo de Datos



LOGICA

El usuario Entre al programa y se abre el menú principal en el que elige una de las seis opciones:

Si elige la opción de “1. Cargar Archivo de entrada”, entrara a un interfaz de Tkinter donde se elige el archivo de texto “.glc” que se utiliza en el programa.

Si elige la opción de “2. Mostrar información general de la gramática”, el programa imprime las gramáticas cargado en memoria y espera que el usuario elige uno. Al elegir una gramática, el programa se escribe en la consola él información general de la gramática.

Si elige la opción de “3. Generar Autómata de Pila Equivalente”, el programa imprime las gramáticas cargado en memoria y espera que el usuario elige uno. Al elegir una gramática, el programa genera una HTML con una gráfica del gramático elegido.

Si elige la opción de “4. Reporte de Recorrido”, el programa imprime las gramáticas cargado en memoria y espera que el usuario elige uno. Al elegir una gramática, el programa pide una cadena. Esta cadena se prueba con la gramática elegido y el programa se genera una HTML donde se muestre varias imágenes con sus transiciones para ver si la cadena es aceptada o no.

Si elige la opción de “5. Desplegar todas a archivo”, el programa imprime las gramáticas cargado en memoria y espera que el usuario elige uno. Al elegir una gramática, el programa pide una cadena. Esta cadena se prueba con la gramática elegido y el programa se genera una HTML donde se muestre una tabla con las transiciones de la cadena para ver si la cadena es aceptada o no.

Si elige la opción de “6. Salir”, el programa sale.

Mientras el Usuario no elige la opción 6, el programa sigue ejecutando y se regresa al menú principal.