

MANUAL TÉCNICO

Universidad San Carlos de Guatemala
Lenguajes Formales y Programación B-
Pablo Josué Barahona Luncey
202109715
3560855890101

PROYECTO 2



Acerca del programa

Este programa fue desarrollado con el lenguaje formal de programación Python, el objetivo del programa fue crear un compilador por medio de parser. Posterior a eso crear una página HTML y CSS con el archivo proporcionado

En este programa se usaron distintas librerías y se realizó con diferentes métodos

La librería principal fue Tkinter, debido a que se tenía que hacer interfaz gráfica

Métodos

Para el menú principal utilizamos la librería Tkinter y también para toda las funcionalidades gráficas del programa.

El programa posee dos archivos, con nombre "Operadora" que es la clase main y "analizador"

Clase Operadora

- **Nuevo()**

Este método guarda lo que se encuentre en el editor de texto y limpia la entrada para un nuevo documento

- **abrirArchivo()**

Se hizo un método para abrir archivo, el cuál se inicia al presionar el botón "Abrir Archivo", se utilizó el importe `filedialog` y `nthpath`

- **guardar()**

Este método guarda el archivo con el mismo nombre del original.

- **guardarcomo()**

Este método guarda el archivo con un nuevo nombre

- **analizar()**

Este método llama a la clase Analizar, la cual se especifica más adelante y también abre el reporte PDF de Analizar

- **errores()**

Este método llama a otro método de la clase Analizar y también abre un archivo PDF generado en la otra clase

Métodos

- salirPro()

Método que sale del programa

- manualusuario()

Este método abre el manual de usuario

- manualtecnico()

Este método abre el manual técnico

- info()

Este método muestra la información del programador

En la interfaz gráfica se muestra un editor de texto, dónde se muestra la información del archivo cargado, también la posición de filas y columnas. Esto fue creado con un evento que se actualiza cuando se escribe o se presiona el cursor.

analizador.py

Este archivo maneja 2 clases llamadas L_tokens y Analizador
Clase L_tokens

Esta clase almacena los tokens para el autómata, acá utilizamos la librería "enum" y su función es especificar los valores que se esperan al leer el archivo de entrada

Clase Analizador

Este es el núcleo del programa.

- `__init__()`

Este método inicia las variables a utilizar

- `quitar()`

Este método quita las líneas leídas

- `aumentarLinea()`

Este método lee las líneas siguientes

- `verificarToken()`

Este método interpreta los tokens por orden y retorna una cadena actualizada con el token ya leído.

- `VerificarID()`

Este método interpreta los tokens por orden, al igual que el método `verificarToken`, con la diferencia de que se tiene un alfabeto para verificar que se reciba correctamente el ID

analizador.py

- Texto()

Este método interpreta los tokens por orden, el método es para la etiqueta Texto, acá se analiza el texto que se otorga en el archivo de entrada

- LecturaporEstados()

Este método es el parser, acá es donde se leen los estados para poder interpretar el archivo xml, este método está basado en un AFD

- compile()

Este método inicia el programa y la clase analizador, empieza leyendo el texto y hacerle arreglos respectivos

- htmlanalizar()

Este HTML es el encargado de generar el HTML de los resultados operativos que se obtienen

- htmlerrores()

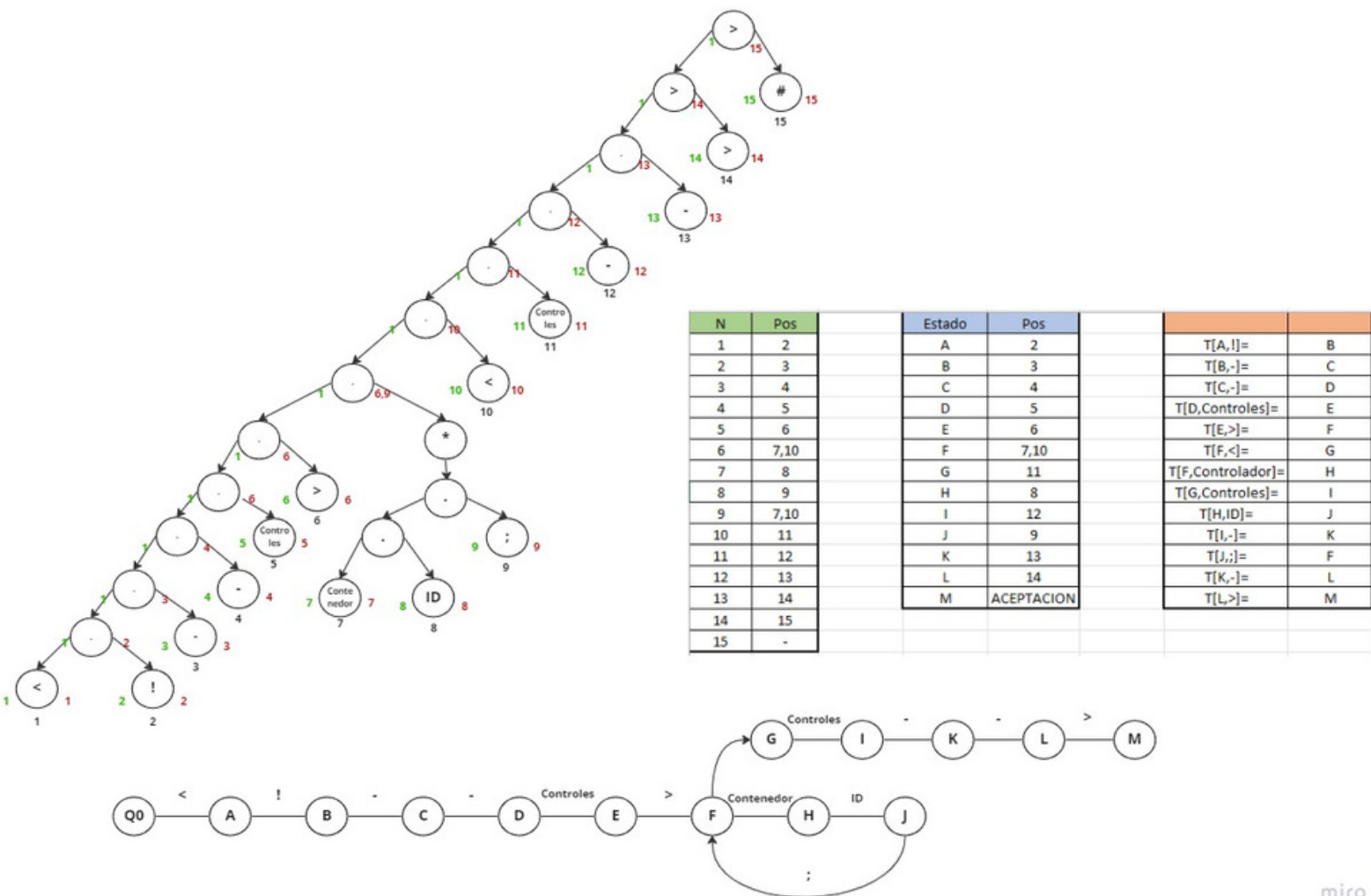
Este método se encarga de generar el HTML de los errores generados

- htmltokens()

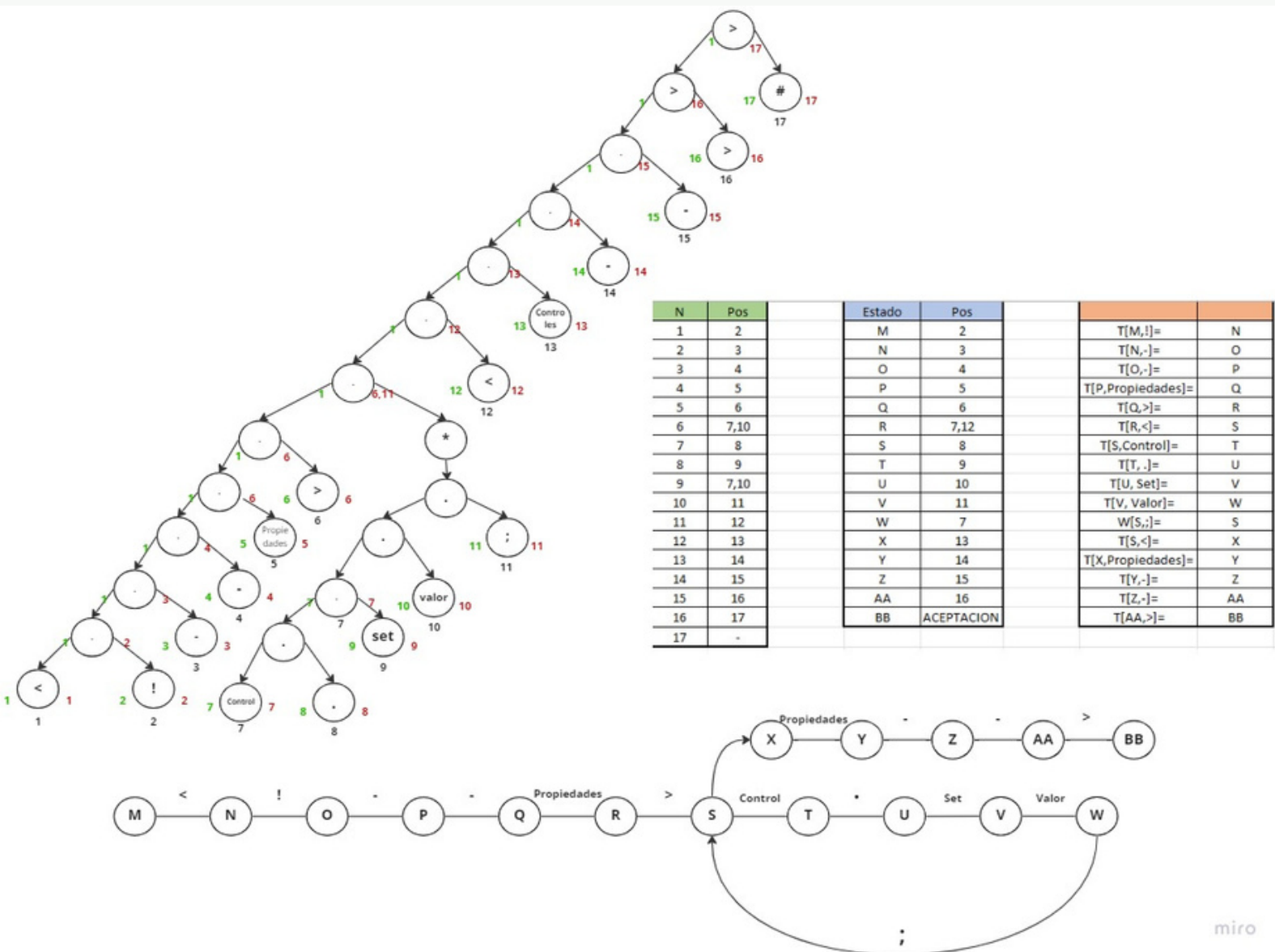
Este método se encarga de generar el HTML de los tokens utilizados para leer el archivo

Diagramas de árbol con AFD

Esta es una muestra gráfica del AFD con el orden de nuestro programa y el uso de los tokens. son 3 en total



Diagramas de árbol con AFD



Diagramas de árbol con AFD

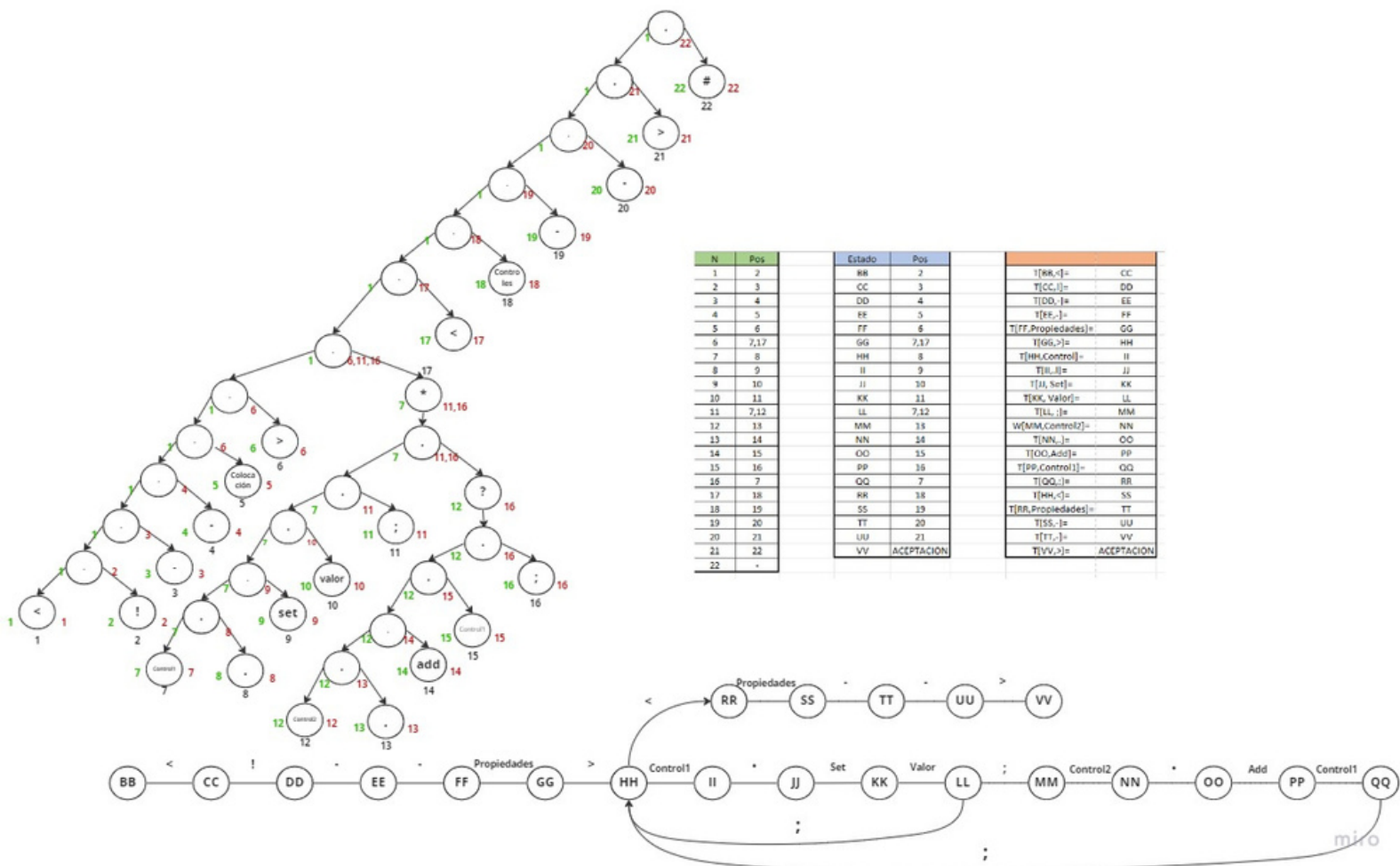
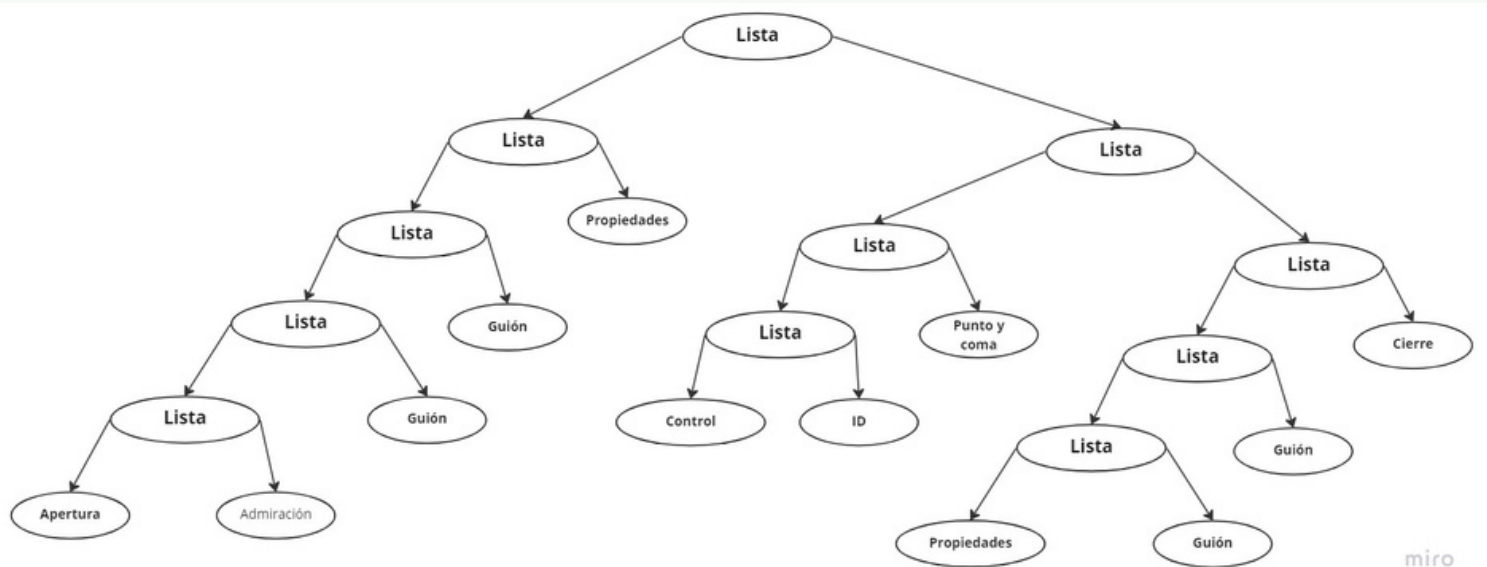


Tabla de tokens

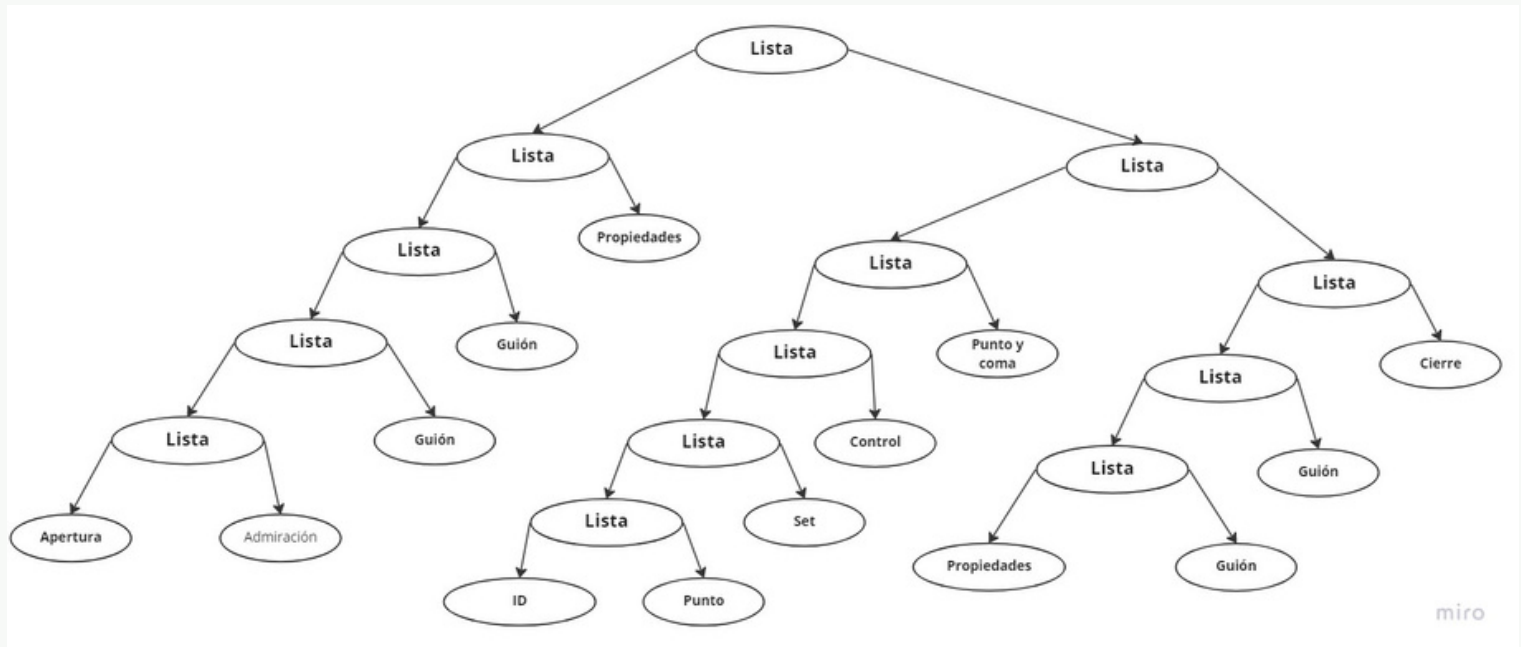
Nº	Token	Lexema
1	APERTURA	<
2	ADMIRACIÓN	!
3	GUIÓN	-
4	CONTROLES Y PROPIEDADES	PROPIEDADES, CONTROLES, COLOCACION
5	CIERRE	>
6	PUNTO	.
7	PUNTO Y COMA	;
8	VALOR	ID proporcionado
9	SET	Diferentes Set
10	ADD	add
11	TIPOS DE CONTROL	Diferentes controles
12	ID	id

Diagramas libre de contexto

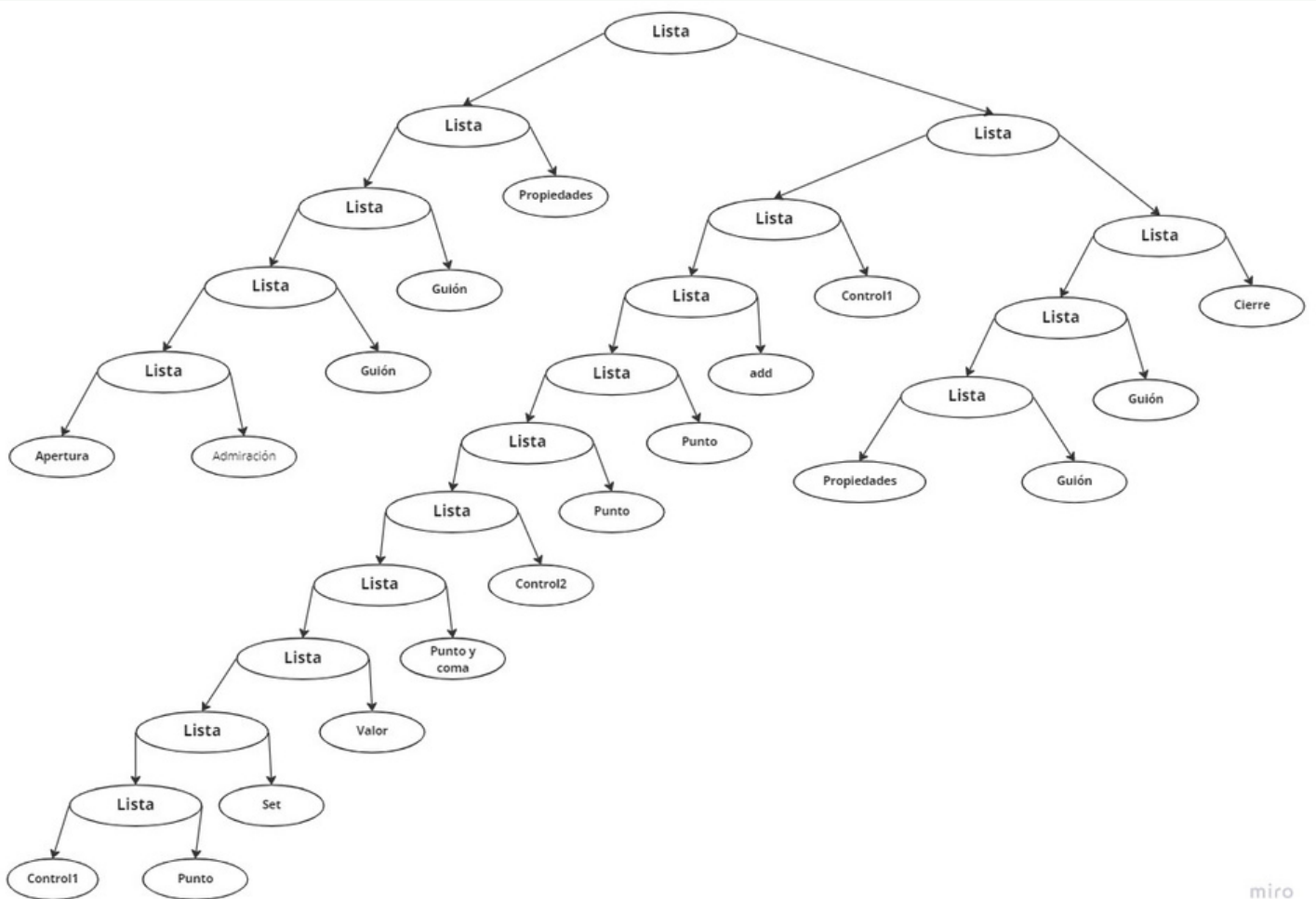


miro

Diagramas libre de contexto



Diagramas libre de contexto



miro