

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA  
CENTRO UNIVERSITARIO DE OCCIDENTE  
DIVISIÒN DE CIENCIAS DE LA INGENIERÍA



MANUAL TÉCNICO DE PRÁCTICA 1  
LABORATORIO DE ORGANIZACIÓN DE LENGUAJES Y COMPILADORES

PRESENTADO POR:

DANIEL EDUARDO BAUTISTA FUENTES

201930588

DOCENTE  
ING. JOSÈ MOISÉS GRANADOS GUEVARA  
AUXILIAR

...

QUETZALTENANGO - QUETZALTENANGO - GUATEMALA

28/02/2022

# INDICE

<b>Información de la aplicación</b>	<b>3</b>
<b>Analizadores</b>	<b>4</b>
Analizador léxico	4
Analizador sintáctico	8
Estados del analizador	8
Producciones del analizador	12
Precedencias	14

# Contenido

## Información de la aplicación

Proyecto escrito con ayuda del IDE Android Studio junto con VSCode y contiene las siguientes características.

1. Lenguajes de programación: Java, Kotlin, JFlex, JCUP
2. Nombre del proyecto: Compi1Prac1
3. Versión del proyecto: 1.0.0-hw

La aplicación está hecha para correr en Android, la versión utilizada durante el desarrollo es Android Nougat 11.0 con API 24. Podemos ejecutar el proyecto usando el IDE de netbeans/intelliJ o instalando el apk en el celular.

# Analizadores

## Analizador léxico

Analizador encargado de leer los TOKENS generados por el lenguaje para graficar.  
Nuestros TOKENS aceptados están definidos mediante la siguiente sintaxis

GRAMATICA	TOKEN GENERADO	DESCRIPCION
,	COMMA	Separador “coma”, para dividir declaraciones de números
:	COLON	Usado para separar declaraciones de atributos
;	SEMI_COLON	Usado para terminar las declaraciones de atributos
{	OPEN_BRACES	Se usa para aperturar la definición de un gráfico o un atributo de unión
}	CLOSE_BRACES	Se usa para clausurar la definición de un gráfico o atributo de unión
[	OPEN_BRACKET	Usado para juntar un grupo de números o valores para una declaración de atributo de gráfica
]	CLOSE_BRACKET	Usado para juntar un grupo de números o valores para una declaración de atributo de gráfica
Regex: [\s\t\r\n]+	whitespace	Todos los espacios, tabulaciones, salto de línea, retornos de valor son ignorados para el analisis lexico y sintactico

Regex: ("#"(.)*)	/* IGNORE */	Los comentarios inician con un # y son ignorados para el análisis
+	ADDITION	Representa una suma
-	SUBTRACTION	Representa una resta
*	MULTIPLICATION	Representa una multiplicación
/	DIVISION	Representa una división
(	OPEN_PARENTHESIS	Usado para unir un conjunto y cálculos matemáticos, también para dividir uniones de atributos de gráfica
=	CLOSE_PARENTHESIS	Usado para unir un conjunto y cálculos matemáticos, también para dividir uniones de atributos de gráfica
Regex: ((D d)ef)	DEF	Denota el inicio de una declaración de gráfica
Barras	GRAPH_BAR	Denota que la gráfica que se está declarando debe ser de barras
Pie	GRAPH_PIE	Denota que la gráfica que se está declarando debe ser de tipo pie
título	TITLE_ATTR	Usado para especificar el título que lleva la gráfica
unir	MERGE_ATTR	Une el valor del eje x en i con el valor de eje y en j. O también puede unir los valores de una etiqueta con un valor para gráficas de Pie.

ejex	X_AXIS_ATTR	Denota que se declararán atributos para el ejex de una gráfica de barras. Este atributo solo puede ser usado si la grafica es de tipo Barra.
ejeY	Y_AXIS_ATTR	Denota que se declararán atributos para el ejeY de una gráfica de barras. Este atributo solo puede ser usado si la grafica es de tipo Barra.
tipo	TYPE_ATTR	Denota que se declarará el atributo Tipo de una gráfica de Pie. Este atributo solo puede ser usado en una gráfica de Pie.
etiquetas	LABEL_ATTR	Denota que se declararán las etiquetas de una gráfica de Pie. Este atributo solo puede ser usado en una gráfica de Pie.
valores	VALUES_ATTR	Denota que se declararán los valores de una gráfica de Pie. Este atributo solo puede ser usado en una gráfica de Pie.
total	TOTAL_ATTR	Denota que se declarará el total que debe sumar los VALUES_ATTR declarados. Este solo debe venir cuando las gráficas sean de tipo Cantidad, de lo contrario, se asume que el total es de 360
extra	EXTRA_ATTR	Denota que se hará una declaración de extra, este extra se muestra en la descripción de la gráfica

Cantidad	TYPE_CUANT_ATTR	Si la gráfica es de tipo Cantidad, debe estar escrito con C mayúscula, este TOKEN, válida eso.
Porcentaje	TYPE_PERC_ATTR	Si la gráfica es de tipo Porcentaje, debe estar escrito con P mayúscula, este TOKEN, válida eso.
Ejecutar	EXECUTE_ACT	Ejecutar significa que el valor que declare, debe ser graficado.
Regex: (\"([a-zA-Z] \\s) [0-9])+\\")	VAL_ON_COMILLAS	Usado para declarar valores de texto a guardar, todo lo que se encuentre entre las comillas es tomado como un único texto
Regex: [0-9]+ = Ints	NUM_INT	Numero entero
Regex: [0-9]+.[0-9]+ = Decimal	NUM_DEC	Numero decimal
Regex: ({Letter}({Letter} {Ints})*)	ID	ID, token usado para identificar cuando una cadena de texto NO coincide con un TOKEN anterior. Así poder comprar si deF es parecido a Def y sugerir la corrección
Regex: [^]	ERROR	Token con lexemas no existentes en el contexto de nuestra aplicación

# Analizador sintáctico

El analizador cuenta con estados terminales y no terminales. Estos estados son los siguientes

## Estados del analizador

NOMBRE	TIPO	DESCRIPCIÓN
COMMA	terminal	Una coma
COLON	terminal	Dos puntos
SEMI_COLON	terminal	Punto y coma
OPEN_BRACES	terminal	Apertura de llaves
CLOSE_BRACES	terminal	Cerradura de llaves
OPEN_BRACKET	terminal	Apertura de corchetes
CLOSE_BRACKET	terminal	Cerradura de corchetes
ADDITION	terminal	Suma
SUBTRACTION	terminal	Resta
MULTIPLICATION	terminal	Multiplicacion
DIVISION	terminal	Division
OPEN_PARENTHESIS	terminal	APertura perentesis
CLOSE_PARENTHESIS	terminal	Cerradura parentesis
DEF	terminal	def
GRAPH_BAR	terminal	Barra
GRAPH_PIE	terminal	Ple
TITLE_ATTR	terminal	titulo
MERGE_ATTR	terminal	unir



X_AXIS_ATTR	terminal	ejex
Y_AXIS_ATTR	terminal	eje y
TYPE_ATTR	terminal	tipo
LABEL_ATTR	terminal	etiquetas
VALUES_ATTR	terminal	valores
TOTAL_ATTR	terminal	total
EXTRA_ATTR	terminal	extra
TYPE_CUANT_ATTR	terminal	Cantidad
TYPE_PERC_ATTR	terminal	Porcentaje
EXECUTE_ACT	terminal	Ejecutar
VAL_ON_COMILLAS	terminal	"texto"
NUM_INT	terminal	1   4 ...
NUM_DEC	terminal	23.123   123.1...
ID	terminal	Texto
ERROR	terminal	Todo lo no reconocido por nuestra gramática
main	no terminal	Estado inicial
bar_dec_sub	no terminal	División de declaración de grafica de barra
pie_dec_sub	no terminal	División de declaración de gráfica de pie
def_graph_section	no terminal	Sección de definición de gráfico
def_graph	no terminal	Definición de gráfico
exec_graph_section	no terminal	Sección de definición de ejecución
exec_graph	no terminal	Declaración de ejecución

		de gráfico
bar_dec	no terminal	Declaración de grafica de barra
pie_dec	no terminal	Declaración de gráfica de pie
title_dec	no terminal	Declaración de atributo título
x_ax_dec	no terminal	Declaración de atributo ejex
y_ax_dec	no terminal	Declaración de atributo ejey
merge_dec	no terminal	Declaración de atributo de union
type_dec	no terminal	Declaración de atributo de tipo
tags_dec	no terminal	Declaración de atributo de etiquetas
vals_dec	no terminal	Declaración de atributo de valores
total_dec	no terminal	Declaración de atributo de total
extra_dec	no terminal	Declaración de atributo extra
mult_comillas	no terminal	Múltiples valores entre comillas
mult_nums	no terminal	Múltiples valores de números
mult_braces	no terminal	Múltiples valores entre llaves
brace_conj	no terminal	Conjunto de dígitos entre llaves

digit	no terminal	Numero
arithm	no terminal	Expresión aritmética
arithm_sim	no terminal	Simbolo aritmetico

## Producciones del analizador

El analizador sintáctico fue configurado usando las siguientes reglas de producción

Origen	Producción
main	def_graph_section exec_graph_section
	def_graph_section
	exec_graph_section
def_graph_section	def_graph_section def_graph
	def_graph
	error
exec_graph_section	exec_graph_section exec_graph
	exec_graph
exec_graph	EXECUTE_ACT OPEN_PARENTHESIS VAL_ON_COMILLAS CLOSE_PARENTHESIS SEMI_COLON
def_graph	DEF GRAPH_BAR OPEN_BRACES bar_dec CLOSE_BRACES
	DEF GRAPH_PIE OPEN_BRACES pie_dec CLOSE_BRACES
bar_dec	bar_dec bar_dec_sub
	bar_dec_sub
bar_dec_sub	title_dec
	x_ax_dec
	y_ax_dec
	merge_dec
	error
pie_dec	pie_dec_sub
	pie_dec pie_dec_sub
pie_dec_sub	title_dec

	type_dec
	tags_dec
	vals_dec
	total_dec
	merge_dec
	extra_dec
	error
title_dec	TITLE_ATTR COLON VAL_ON_COMILLAS SEMI_COLON
x_ax_dec	X_AXIS_ATTR COLON OPEN_BRACKETS mult_comillas CLOSE_BRACKETS SEMI_COLON
y_ax_dec	Y_AXIS_ATTR COLON OPEN_BRACKETS mult_nums CLOSE_BRACKETS SEMI_COLON
merge_dec	MERGE_ATTR COLON OPEN_BRACKETS mult_braces CLOSE_BRACKETS SEMI_COLON
type_dec	TYPE_ATTR COLON TYPE_CUANT_ATTR SEMI_COLON
	TYPE_ATTR COLON TYPE_PERC_ATTR SEMI_COLON
tags_dec	LABEL_ATTR COLON OPEN_BRACKETS mult_comillas CLOSE_BRACKETS SEMI_COLON
vals_dec	VALUES_ATTR COLON OPEN_BRACKETS mult_nums CLOSE_BRACKETS SEMI_COLON
total_dec	TOTAL_ATTR COLON NUM_INT SEMI_COLON
	TOTAL_ATTR COLON NUM_DEC SEMI_COLON
extra_dec	EXTRA_ATTR COLON VAL_ON_COMILLAS SEMI_COLON
mult_comillas	VAL_ON_COMILLAS
	mult_comillas COMMA VAL_ON_COMILLAS
mult_nums	arithm
	mult_nums COMMA airthm

mult_braces	mult_braces COMMA brace_conj
	brace_conj
brace_conj	OPEN_BRACES NUM_INT COMMA NUM_INT CLOSE_BRACES
arithm	digit
	OPEN_PARENTHESIS arithm CLOSE_PARENTHESIS
	arithm arithm_sim arithm
arithm_sim	ADDITION
	SUBTRACTION
	MULTIPLICATION
	DIVISION
digit	NUM_INT
	NUM_DEC

## Precedencias

EL analizador también hace uso de precedencia para poder resolver problemas algebraicos y de lógica interna. Estos están organizados de la siguiente forma

1. Suma y resta tienen prioridad 1 por la izquierda
2. Multiplication y división tienen prioridad 2 por la izquierda
3. Apertura y cerradura de paréntesis tienen prioridad 3 por la izquierda y se usan solamente para agrupación