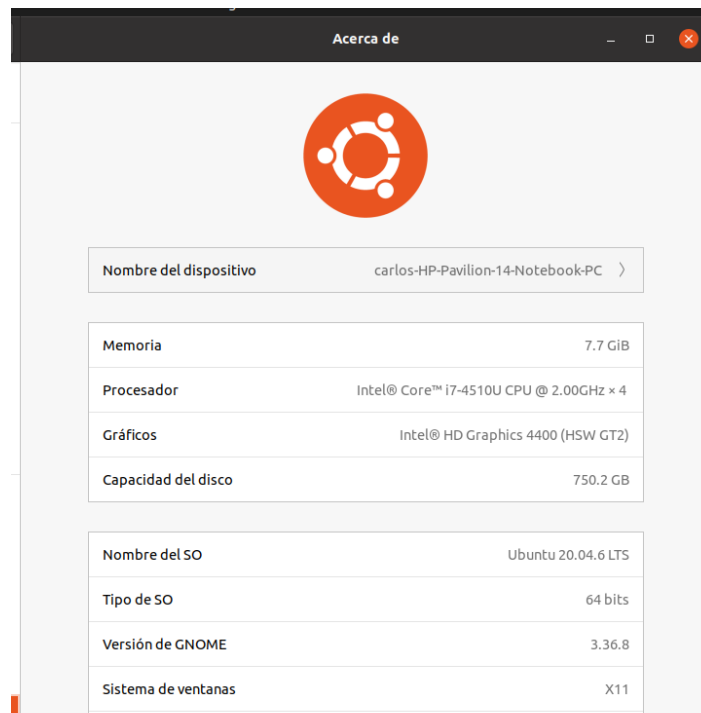


Manual técnico

para la solución de los requerimientos del enunciado se utilizaron las siguientes tecnologías:

- el programa esta desarrollado en java
- para la GUI se utilizo java swing
- versión de librería de jflex es: jflex-full-1.8.2.jar
- versión de librería de jcup es: java-cup-11b.jar y java-cup-11b-runtime.jar

entorno de desarrollo



Lógica de la solución

en base a los requerimientos generamos el lenguaje formal que se necesita para el analizador léxico. Definimos los tokens posibles y las ER para los ID y los números enteros de la siguiente manera:

ER:

ENTERO=[0-9]+

ID = [A-Za-z][A-Za-z0-9_]*

LEXEMA

TOKEN

"("
 ")"

PARIZQ
 PARDER

"_"	MENOS
"+"	MAS
"*"	POR
"/"	DIV
","	COMA
"azul"	AZUL
"rojo"	ROJO
"amarillo"	AMARILLO
"verde"	VERDE
"negro"	NEGRO
"morado"	MORADO
"cafe"	CAFE
"rosa"	ROSA
"celeste"	CELESTE
"linea"	LINEA
"curva"	CURVA
"graficar"	GRAFICAR
"circulo"	CIRCULO
"cuadrado"	CUADRADO
"rectangulo"	RECTANGULO
"poligono"	POLIGONO
"animar"	ANIMAR
"objeto"	OBJETO
"anterior"	ANTERIOR

en cuanto al analizador sintáctico se definió la gramática regular de la siguiente manera:

TERMINALES: PARIZQ, PARDER MENOS, MAS, POR, DIV, COMA, AZUL, ROJO, AMARILLO, VERDE, NEGRO, MORADO, CAFE, ROSA , CELESTE, LINEA, CURVA, GRAFICAR, CIRCULO, CUADRADO, RECTANGULO, POLIGONO, ANIMAR, OBJETO, ANTERIOR, ENTERO, ID.

NO TERMINALES

- ini
- instrucciones
- instruccion
- expresion
- graficar
- animar
- color
- tipo
- forma

INICIO: ini.

PRODUCCIONES:

ini::=instrucciones;

instrucciones ::=

```

instruccion
| error instrucciones
;

instruccion ::= graficar | animar;

graficar ::= GRAFICAR forma;

forma ::= CIRCULO PARIZQ ID COMA expresion COMA expresion COMA expresion COMA color
PARDER
| CUADRADO PARIZQ ID COMA expresion COMA expresion COMA expresion COMA color
PARDER
| RECTANGULO PARIZQ ID COMA expresion COMA expresion COMA expresion COMA
expresion COMA color PARDER
| LINEA PARIZQ ID COMA expresion COMA expresion COMA expresion COMA expresion COMA
color PARDER
| POLIGONO PARIZQ ID COMA expresion COMA expresion COMA expresion COMA expresion
COMA expresion COMA color PARDER
;

expresion ::=
    expresion:a MAS      expresion:b    {: RESULT= (Integer.valueOf((String)a) +
Integer.valueOf((String)b) );;}
| expresion:a POR      expresion:b    {:
RESULT=(Integer.valueOf((String)a)*Integer.valueOf((String)b) );;}
| expresion:a MENOS    expresion:b    {: RESULT=(Integer.valueOf((String)a) -
Integer.valueOf((String)b) );;}
| expresion:a DIV      expresion:b    {:
RESULT=(Integer.valueOf((String)a)/Integer.valueOf((String)b) );;}
| ENTERO:a              {:RESULT=Integer.valueOf((String)a);;}
| PARIZQ expresion:a PARDER          {:RESULT=a;;}
;

color ::= AZUL|ROJO|AMARILLO|VERDE|NEGRO|MORADO|CAFE|ROSA|CELESTE;

animar ::= ANIMAR OBJETO ANTERIOR PARIZQ tipo COMA expresion COMA expresion COMA
expresion PARDER;

tipo ::= CURVA | LINEA;

```