Code to Paint

Proyecto final Diseño de compiladores.

6 de abr. de 17

David Eusebio Cazares Pérez

Visión del proyecto.

El objetivo principal del proyecto es desarrollar, por medio de herramientas y recursos existentes, un lenguaje de programación que permita de manera sencilla dibujar figuras y trazos en la pantalla de la computadora.

La categoría del proyecto entra en la sección de un programa para ayudar a jóvenes a desarrollar un entendimiento básico de programación de manera divertida por medio de comandos sencillos. Dichos comandos se utilizarán para la creación de un dibujo que se presentará en la pantalla de la computadora. Mediante comandos sencillos y básicos, como mover a la derecha cierta cantidad de distancia en la pantalla, formarán diferentes trazos que conformarán al final un dibujo.

El proyecto utilizará la librería de “Turtle graphics” para dibujar sobre la pantalla. Se usará ply para la creación de reglas sintácticas y gramaticales.

Las palabras reservadas que se utilizarán en el proyecto son las siguientes:

PROGRAM ::= 'program'

MAIN ::= 'main'

END ::= 'end'

FUNCTION ::= 'function'

RETURN ::= 'return'

IF ::= 'if'

ELSE ::= 'else'

WHILE ::= 'while'

DO ::= 'do'

INT ::= 'int'

DOUBLE ::= 'double'

BOOLEAN ::= 'boolean'

PENUP ::= 'penUp'

PENDOWN ::= 'penDown'

PENCOLOR ::= 'penColor'

PENSIZE ::= 'penSize'

SETX ::= 'setX'

SETY ::= 'setY'

CLEAR ::= 'clear'

MOVEONX ::= 'moveOnX'

MOVEONY ::= 'moveOnY'

ROTATETORIGHT ::= 'rotateToRight'

ROTATETOLEFT ::= 'rotateToLeft'

RECTANGLE ::= 'rectangle'

TRIANGLE ::= 'triangle'

CIRCLE ::= 'circle'

Las tokens a utilizar son los siguientes:

PLUS = '\+'

MINUS = '-'

MULT = '\\*'

DIVIDE = '/'

LPAREN = '\('

RPAREN = '\)'

LBRACE = '\{'

RBRACE = '\}'

LBRACKET = '\['

RBRACKET = '\]'

EQUALS = '=='

ASSIGN = '='

LESS = '<'

GREATER = '>'

SEMICOLON = ';'

COMMA = ','

AND = '&'

OR = '\|'

COMMENT = '\#.\*(\n | $) '

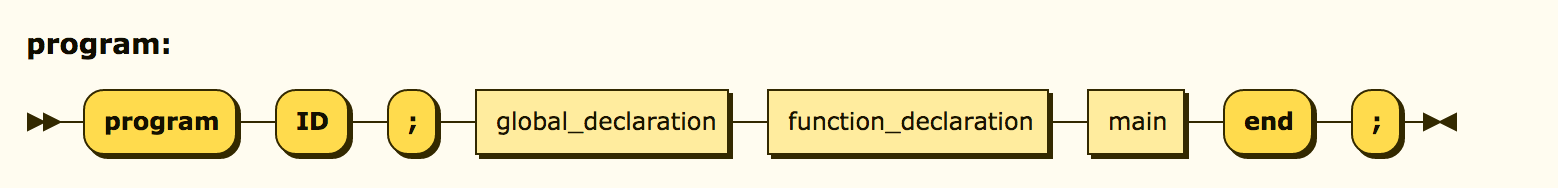
CONST\_INT = '-?\d+'

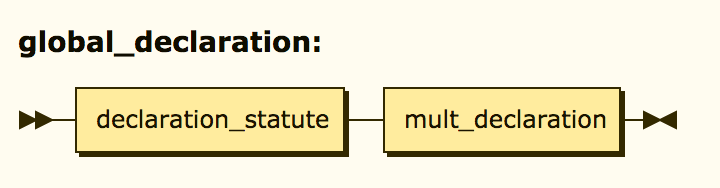
CONST\_DOUBLE = '-?\d+\.\d+'

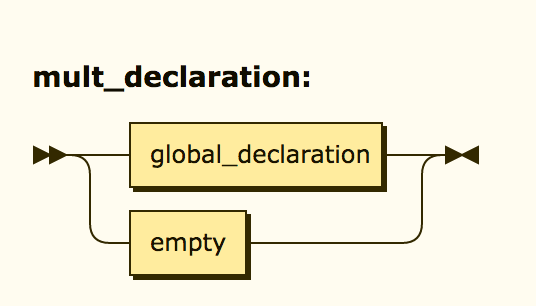
CONST\_BOOLEAN = '(True|False)'

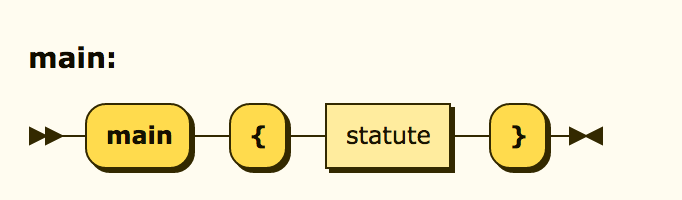
ID = '[a-zA-Z\_][a-zA-Z\_0-9]\*'

**Program**

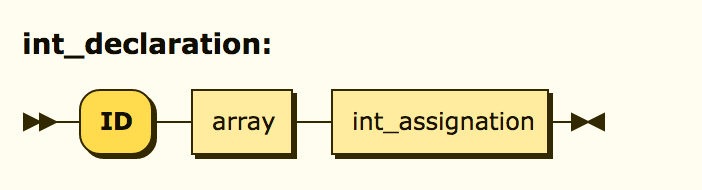


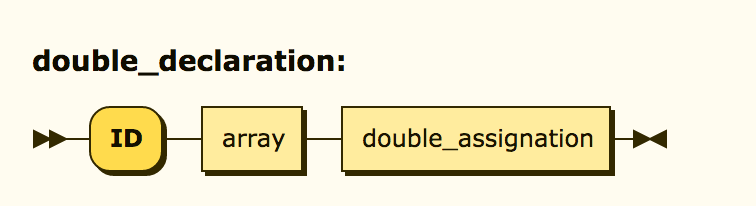


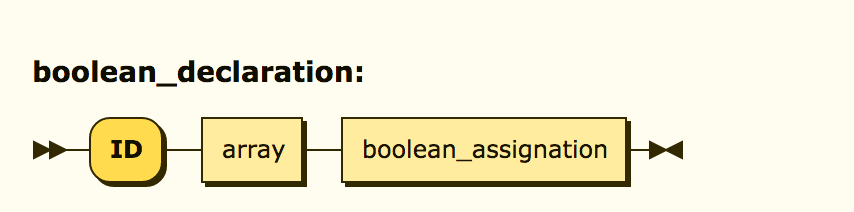




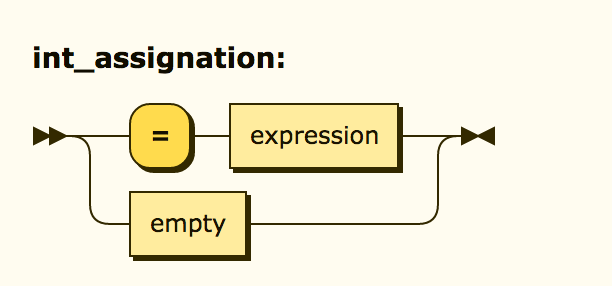
**Declaration**

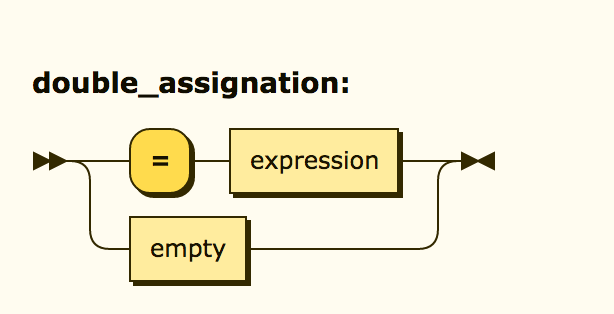
****

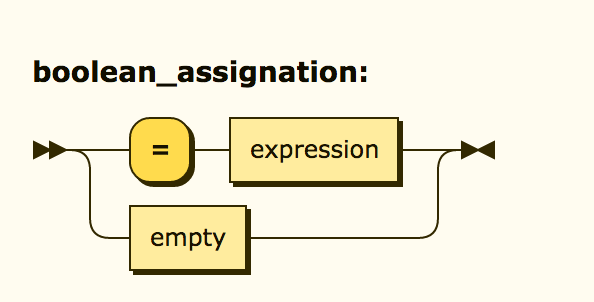


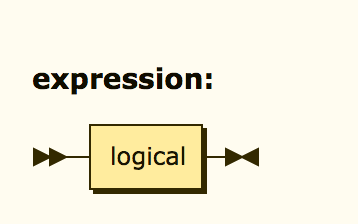


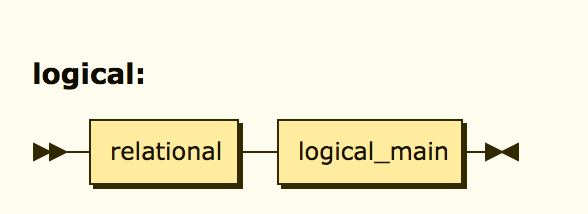
**Assignations**

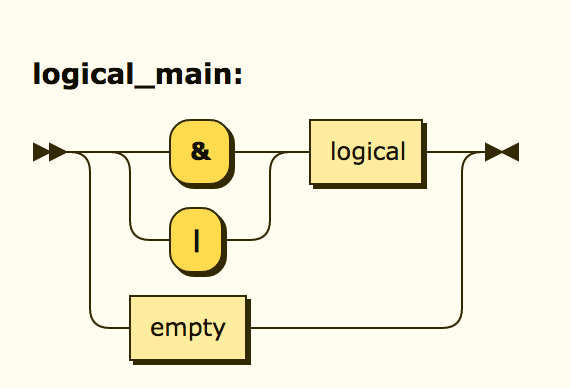


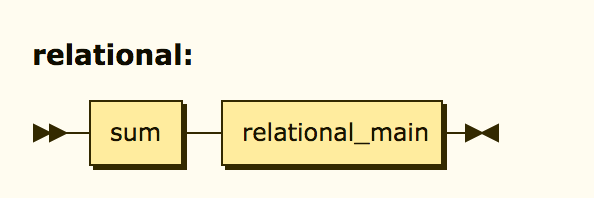


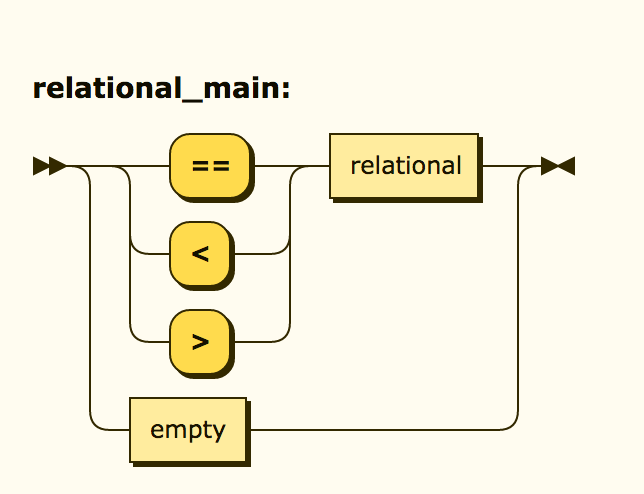


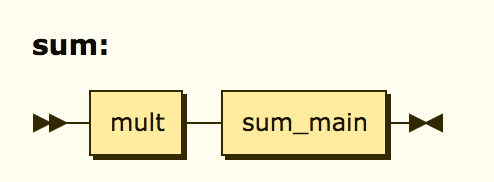
**Expressions**

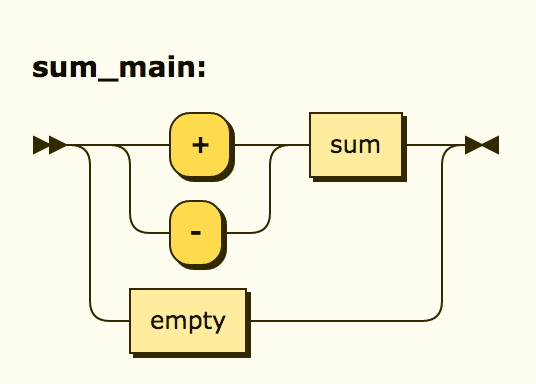


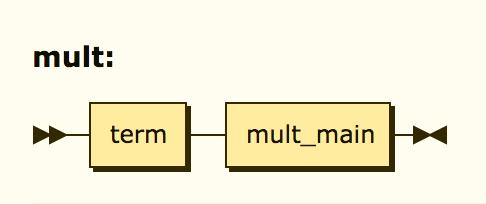


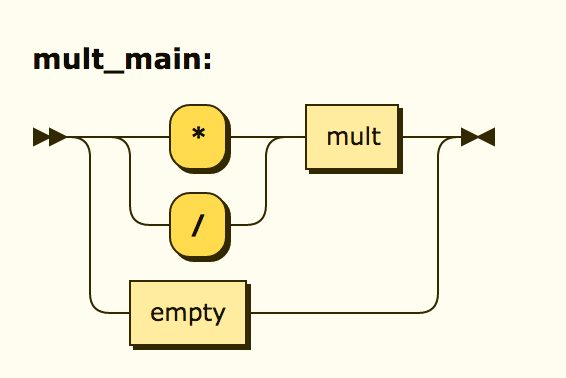




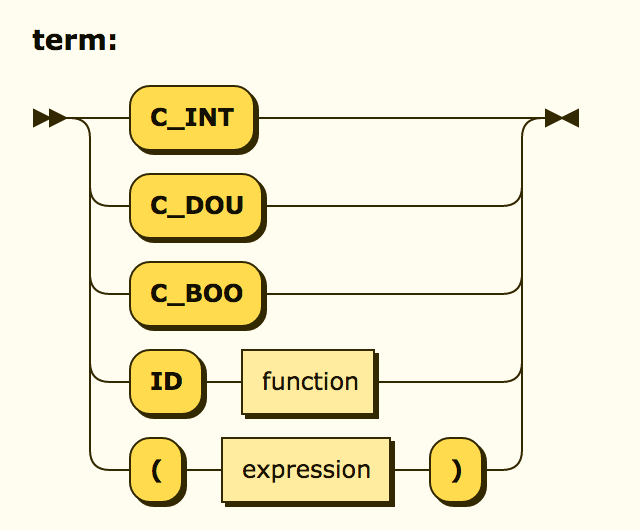




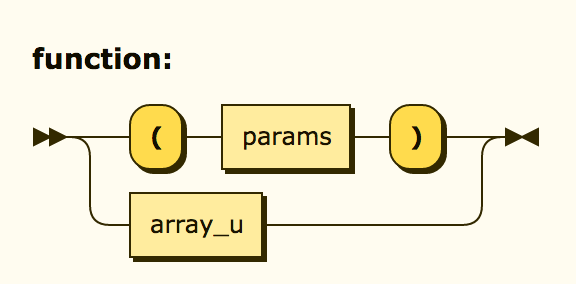


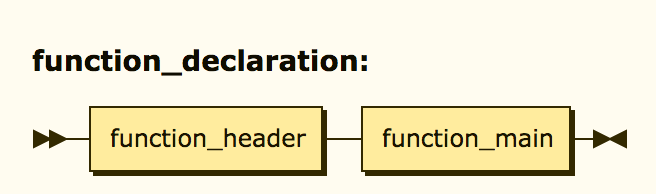


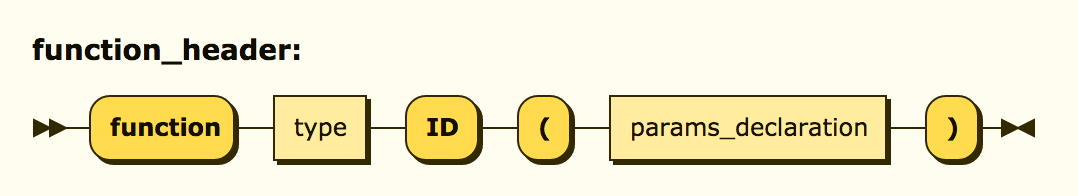
**Term**

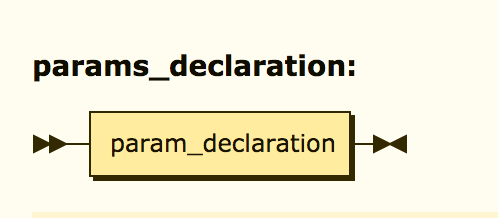
****

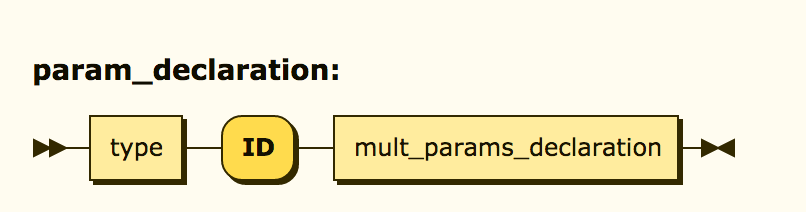
**Function**

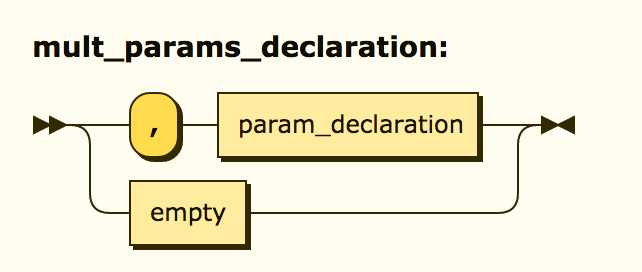
****

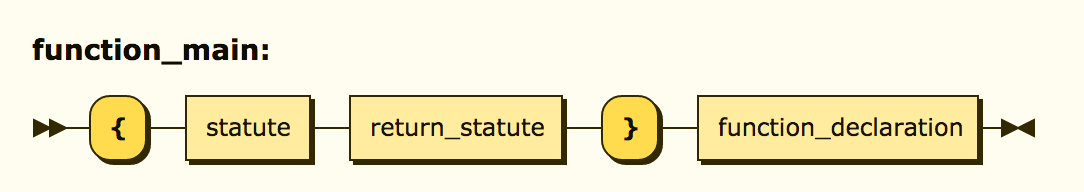




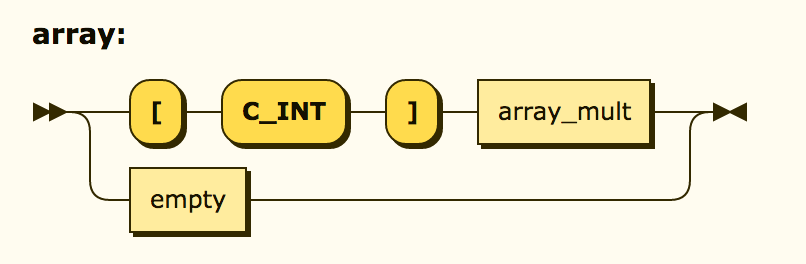


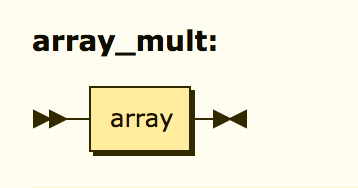


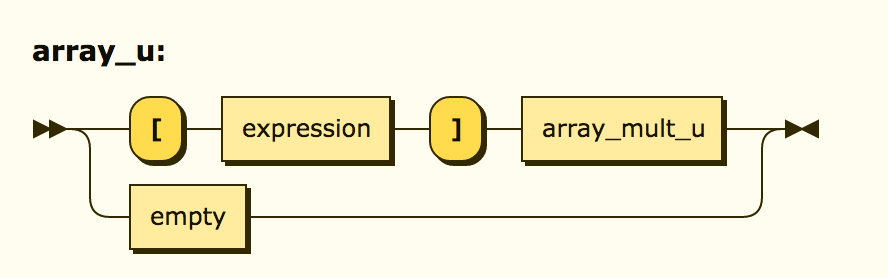


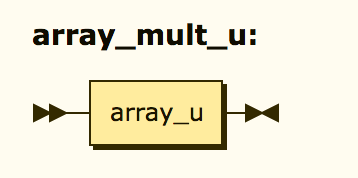


**Array**

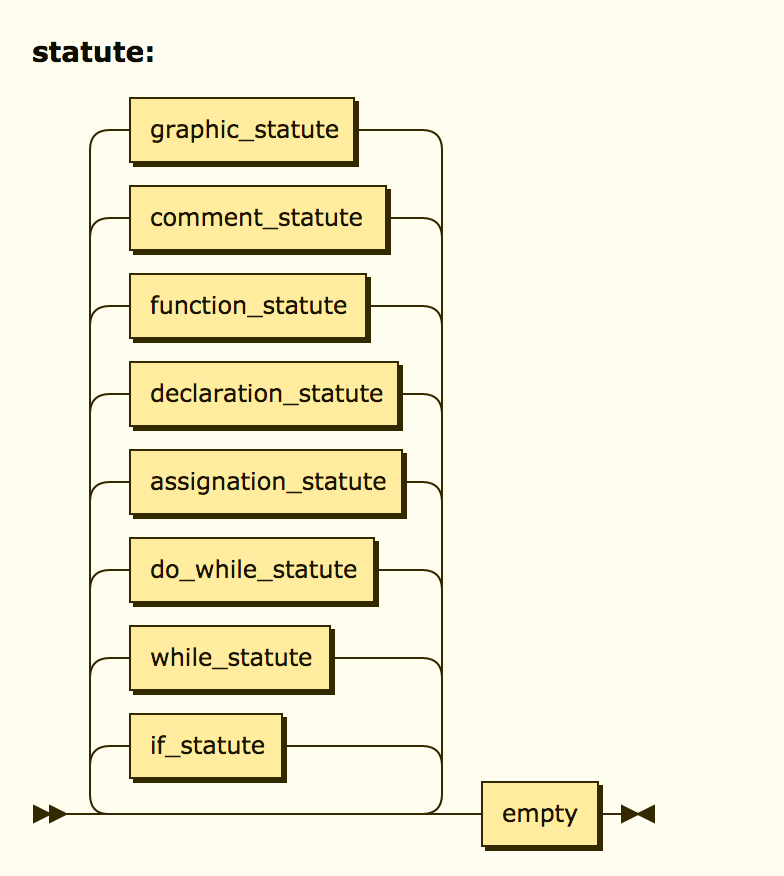


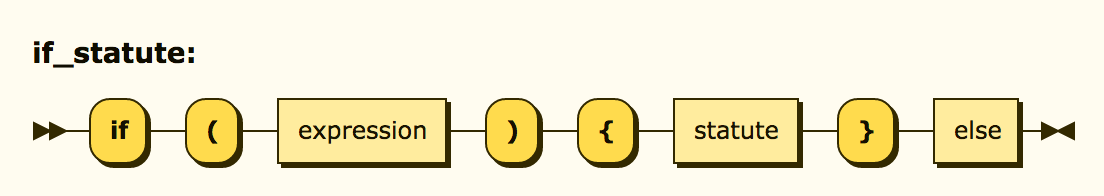


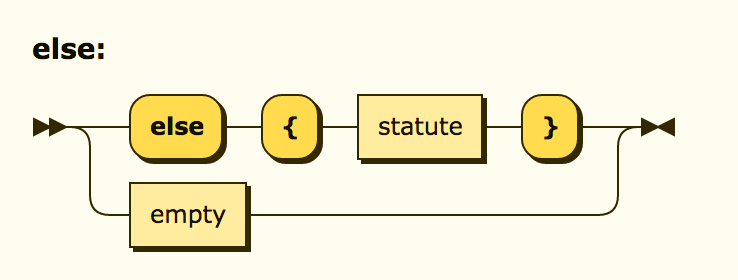


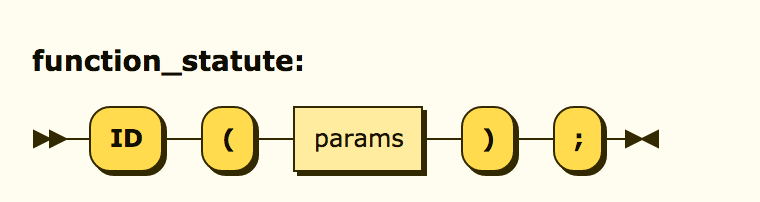


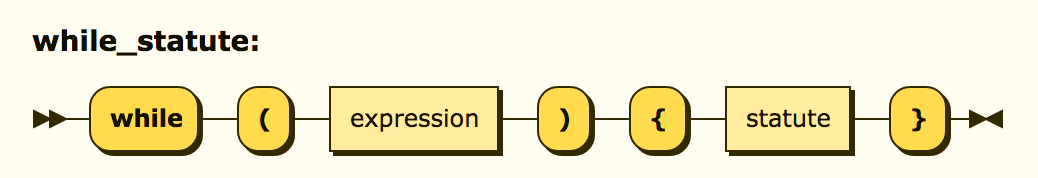
**Statutes**

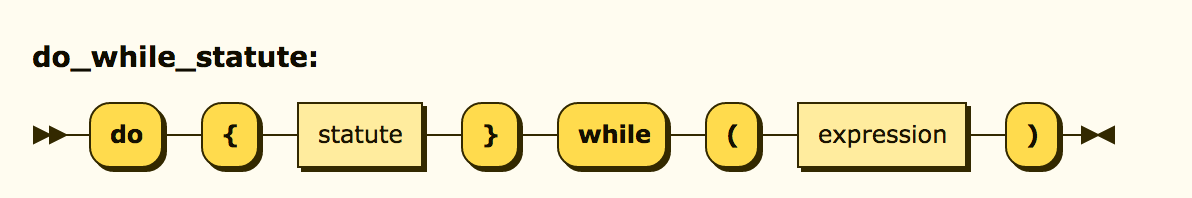
****

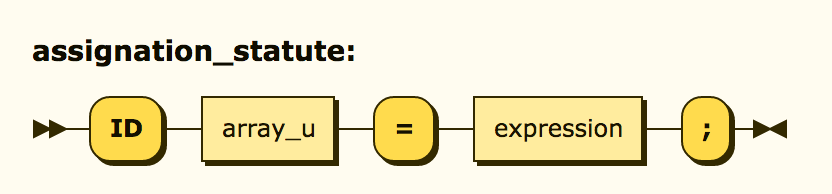


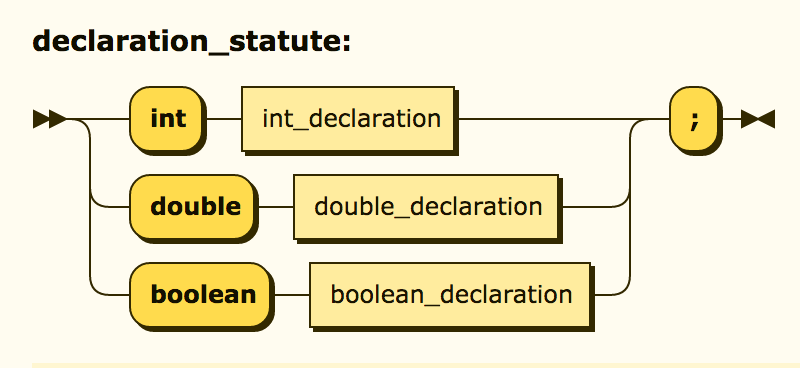


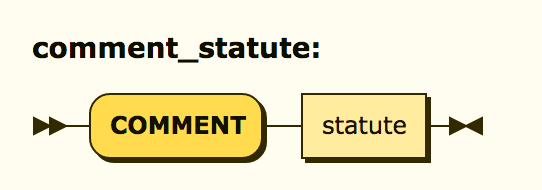


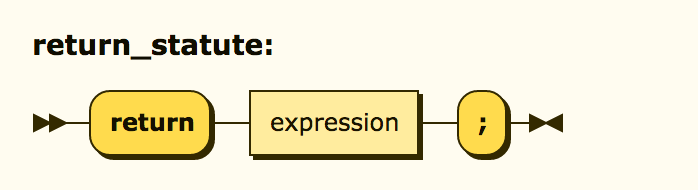




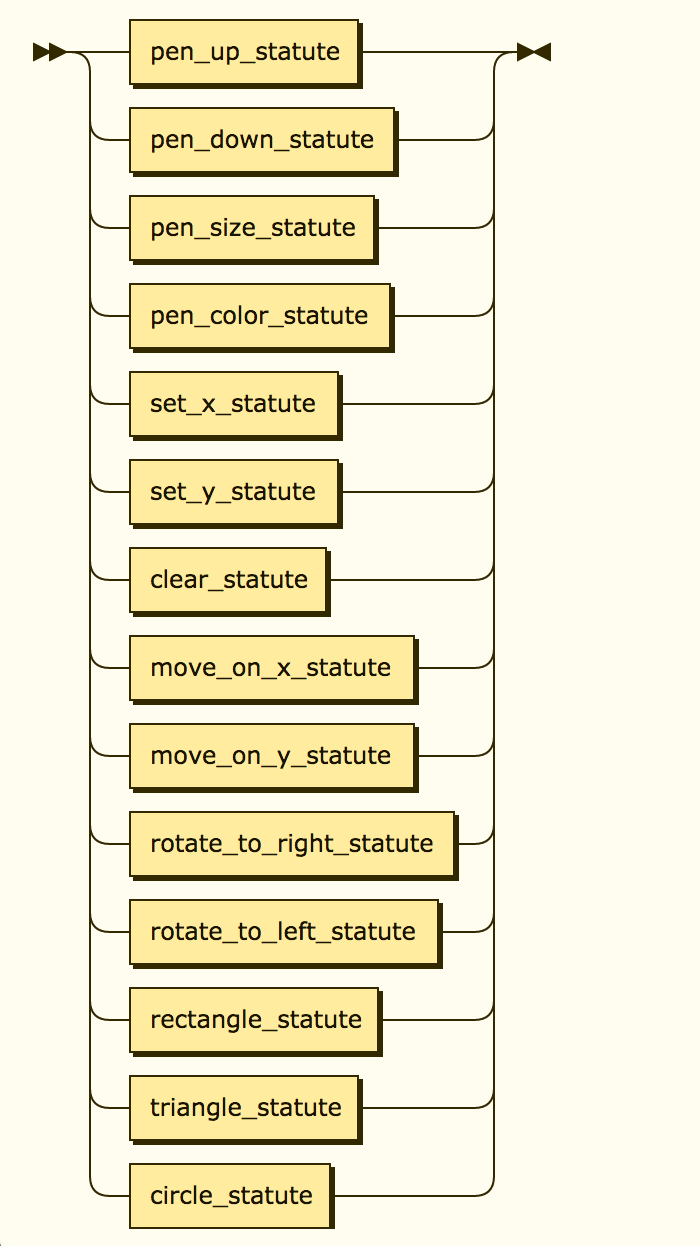


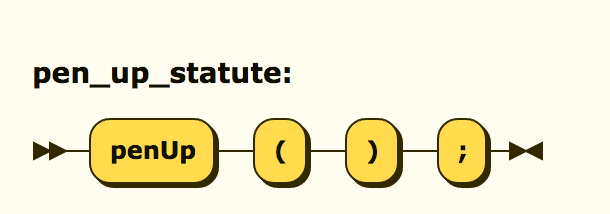


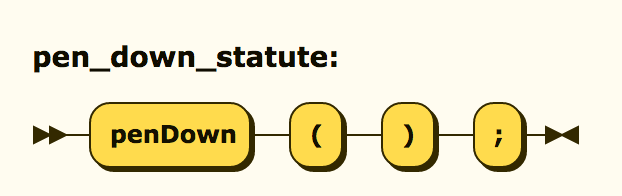


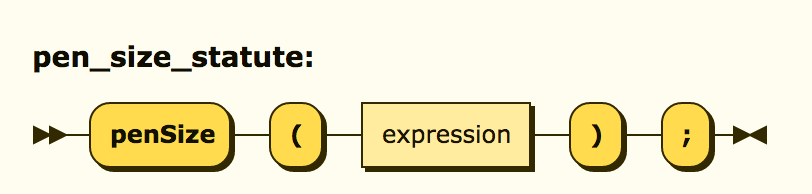


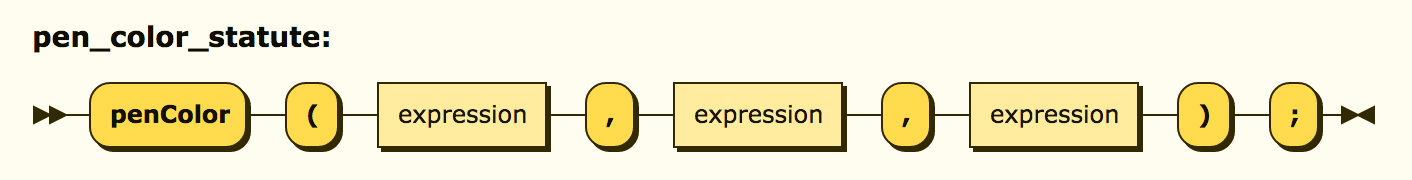
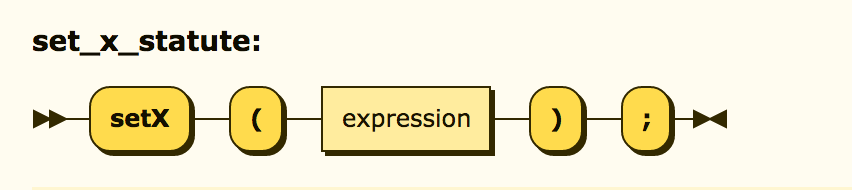
**Graphic statute**

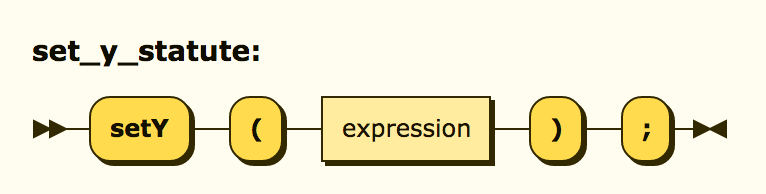


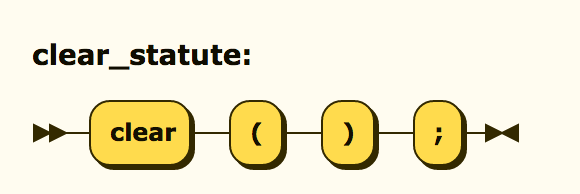


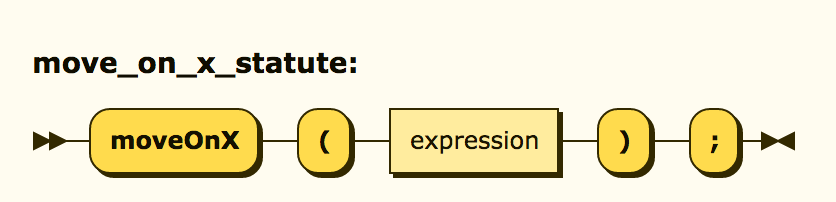


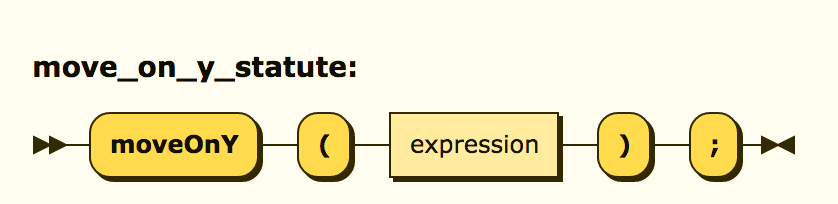


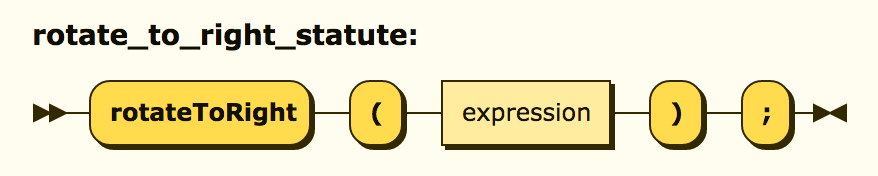


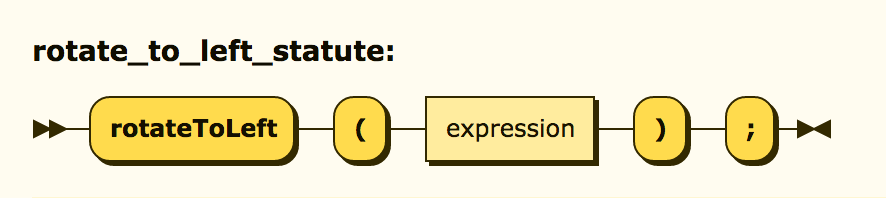


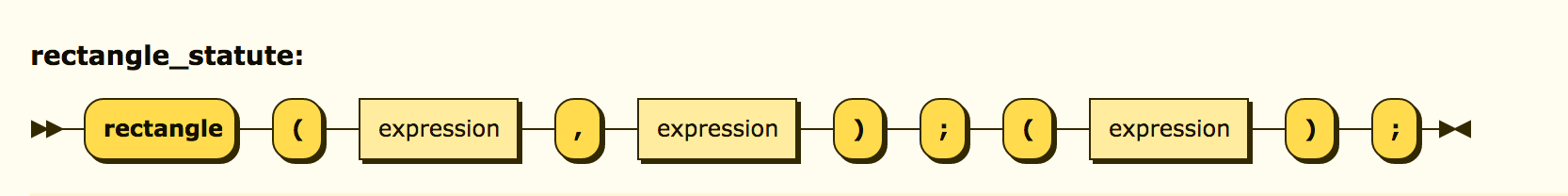


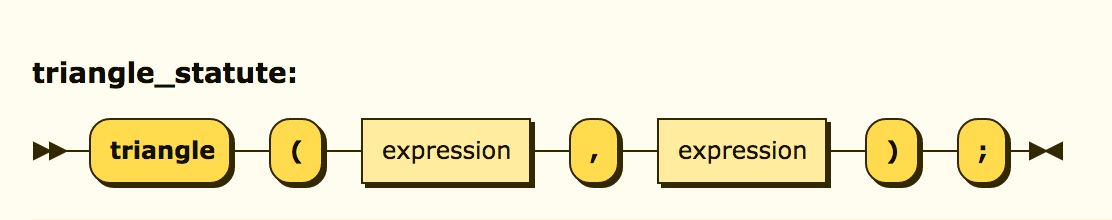


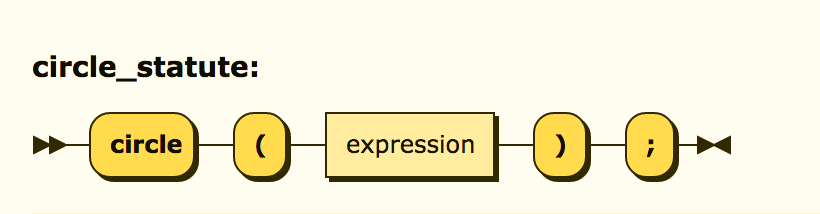












El proyecto contará con las características principales de la semántica básica es decir:

* El chequeo de posicionamiento dentro de un arreglo.
* La declaración e implementación de una función antes de su uso, en otras palabras antes del main.
* Todas las funciones deben tener un valor de retorno.
* Nombre único para variables y funciones, es decir una variable no puede repetir su nombre en otra variable dentro del mismo contexto, pero una variable y función pueden compartir nombre.
* La declaración de variables globales y locales.
* Los enteros y dobles pueden usarse para lógica relacional y en dicho caso se tomará el valor positivo como Verdadero y negativo o cero como falso. No puede usarse otro tipo de funciones como suma o divisiones entre los tipos mencionados anteriormente.
* A pesar de poderse usar para relaciones los valores diferentes a una constante booleana no pueden ser usados para asignar a una variable de tipo booleana.
* Las variables solo pueden contener un tipo y no pueden ser re-declaradas.
* Los arreglos pueden ser usados con variables para acceder a sus campos pero deben ser declarados con una constante entera.
* Los arreglos tienen sus límites desde 0 hasta el número que se indique y se accederán de dicha manera.
* Un número doble debe empezar con por lo menos un dígito antes del punto decimal y debe llevar por lo menos un dígito después del mismo.

Las funciones especiales que se usarán serán para manejar y hacer uso de la librería de Turtle graphics. Las funciones especiales son las especificadas en la sección de estatutos gráficos. Las cuales son :

PenUp: función utilizada para poder mover el apuntador de la pantalla libremente sin dibujar sobre la misma.

PenDown: función usada para comenzar a dibujar. Si esta función se llamó cualquier movimiento por medio de MoveOnX o MoveOnY generará un trazo sobre la pantalla.

PenSize: Define el tamaño del “pincel” para dibujar en la pantalla. Su único argumento define el tamaño.

PenColor: Define el color para pintar sobre la pantalla. Recibe tres argumentos para poner un color en modo RGB.

SetX: Se utiliza para posicionar el cursos de dibujo en un punto específico de la pantalla. Su argumente posiciona al cursor en un punto de la pantalla en el eje de X.

SetY: Se utiliza para posicionar el cursos de dibujo en un punto específico de la pantalla. Su argumente posiciona al cursor en un punto de la pantalla en el eje de Y.

Clear: función para borrar lo que se encuentra dibujado en la pantalla en el momento que se llamó.

MoveOnX: mueve el cursor una cierta cantidad de unidades sobre el eje X. Si el apuntador está en el estado “down” se dibujará sobre la pantalla. El valor puede ser negativo.

MoveOnY: mueve el cursor una cierta cantidad de unidades sobre el eje Y. Si el apuntador está en el estado “down” se dibujará sobre la pantalla. El valor puede ser negativo.

RotateOnX: rota cierta cantidad de grados el cursor hacia la derecha.

RotateOnY: rota cierta cantidad de grados el cursor hacia la izquierda.

Rectangle: se dibuja un rectángulo con el alto y ancho proporcionados empezando en el punto donde se encuentre el cursor. Esta función dibujará sobre la pantalla independiente del estado del cursor.

Triangle: se dibuja un triángulo con el tamaño de la base y la altura proporcionados empezando en el punto donde se encuentre el cursor. Esta función dibujará sobre la pantalla independiente del estado del cursor. El triángulo siempre tendrá al altura máxima en el punto medio de la base.

Circle: se dibuja un circulo con el diámetro proporcionado empezando en el punto donde se encuentre el cursor. Esta función dibujará sobre la pantalla independiente del estado del cursor.

El programa contará con 3 diferentes tipos de datos.

Enteros: que se comprenden como números sin parte decimal.

Dobles: números con parte decimal.

Booleanos: tipos que solo pueden tener dos posibles valores, True y False.

Aparte de los tipos especificados se cuenta con arreglos de dichos tipos.

El proyecto se llevará a cabo mediante el lenguaje de Python y se utilizarán las librerías de Turtle graphic para dibujar en la pantalla y PLY para el manejo de reglas.

Se utilizará la plataforma de GitHub para el manejo de versiones y en el mismo solo participará una persona mediante su computadora personal.