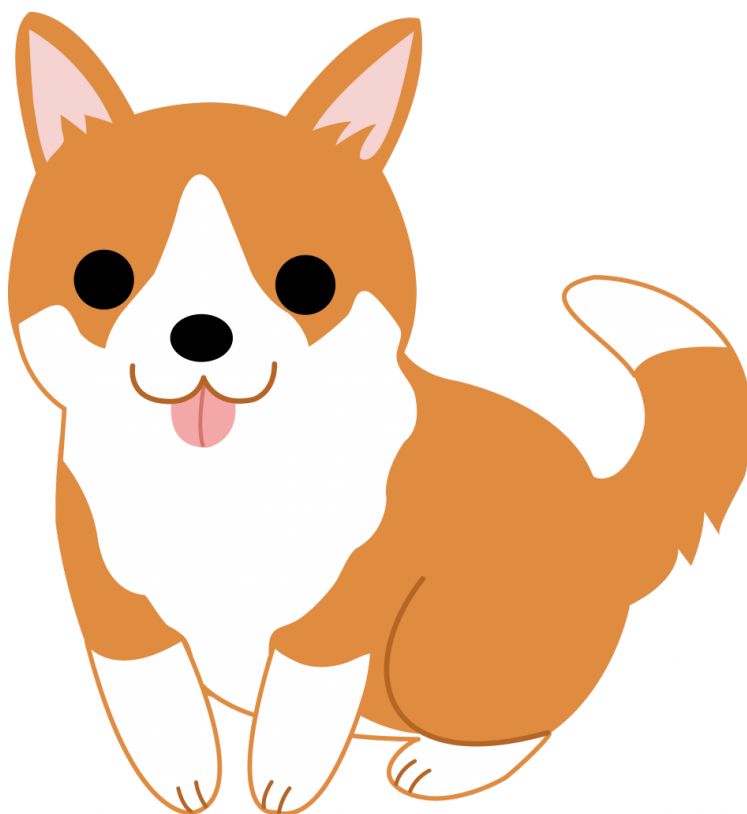




Tecnológico de Monterrey

Compilador
'Baky'



Daniel Treviño Quiroga	A01283132
José Ernesto García Torales	A01365919

Diseño de Compiladores
Ago - Dic 2022

Propósito del Proyecto

El propósito del presente proyecto es crear un lenguaje de programación nuevo y su respectivo compilador con la finalidad de poner en práctica los conocimientos adquiridos durante nuestra carrera como Ingenieros en Tecnologías Computacionales, así como también reforzar todos los conceptos vistos en la clase de Diseño de Compiladores. De igual manera se busca crear un entorno de desarrollo integrado (IDE) para dispositivos móviles en el que se pueda crear, editar, compilar y correr programas escritos en este nuevo lenguaje de programación.

Objetivo del Lenguaje

El principal objetivo de este lenguaje es que sea fácil de aprender, muy intuitivo y que se pueda utilizar en un dispositivo móvil con la ayuda del IDE creado por nosotros. Debe de igual manera contener los elementos básicos de cualquier lenguaje de programación como variables, arreglos, ciclos, condicionales y funciones. La idea general de Baky es que niños pequeños o jóvenes que estén interesados en la programación tengan la oportunidad de poder aprender un lenguaje imperativo básico con la ayuda de un dispositivo móvil.

Requerimientos del Lenguaje

Elementos Básicos

Palabras Clave

Baky al igual que cualquier otro lenguaje de programación utiliza ciertas palabras reservadas para poder identificar ciertas estructuras o instrucciones a ejecutar. Las palabras reservadas en Baky son:

- Baky
 - Esta palabra se utiliza al inicio de todos los programas para identificar que está escrito en Baky, de igual forma se utiliza para nombrar a la función principal del código.
- var
 - Nos indica la declaración de una o más variables.
- int, double, string, char, boolean
 - Se utilizan para declarar el tipo de variable.
- function
 - Nos indica la declaración de una función.
- void
 - Se usa para declarar funciones sin valores de retorno.
- return
 - Se usa para regresar el valor de una función.
- read
 - Se utiliza para leer valores de entrada.
- write
 - Se utiliza para desplegar valores en la terminal.
- if, else, while, from, to, do
 - Se utilizan para condicionales y ciclos.

ID's

Los id's en Baky como en cualquier lenguaje de programación se utilizan para almacenar y reconocer valores en el programa. Todos los id's en Baky deben comenzar con una letra y después aceptan cualquier letra, número o guión bajo. La siguiente expresión regular representa cualquier identificador válido, siempre y cuando no sea una palabra reservada

`[a-zA-z]\w*`

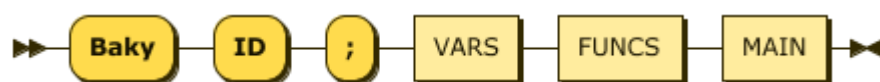
Símbolos

Baky se apoya de varios símbolos especiales, que tienen un uso específico dentro del lenguaje de programación. Los símbolos especiales en Baky son los siguientes:

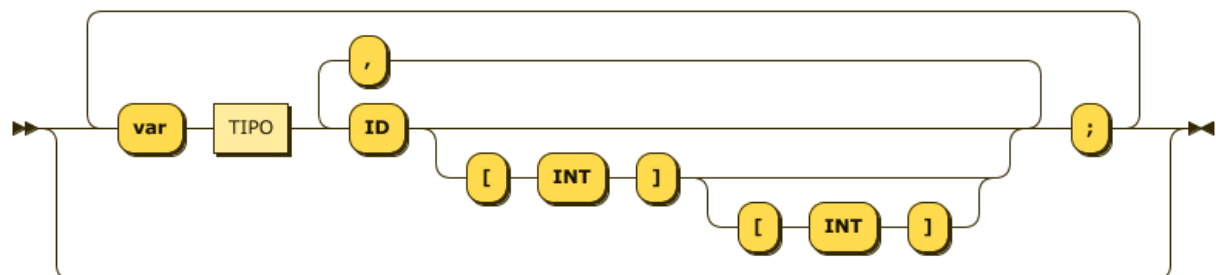
- ;
 - Define el fin de un estatuto.
- ,
 - Separar estatutos.
- []
 - Se usa para referenciar posiciones en arreglos.
- ()
 - Se usa para delimitar los parámetros en llamadas a funciones.
- { }
 - Se usa para delimitar bloques de código.
- =
 - Operador de asignación en Baky.
- || &&
 - Operadores lógicos en Baky.
- + - * /
 - Operadores aritméticos en Baky.
- == != <= >= > <
 - Operadores relacionales en Baky.

Diagramas de sintaxis

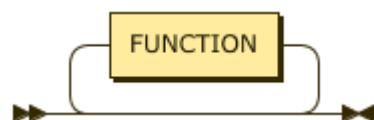
BAKY



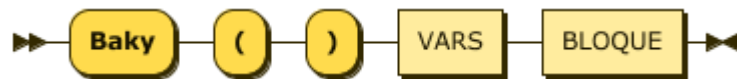
VARS



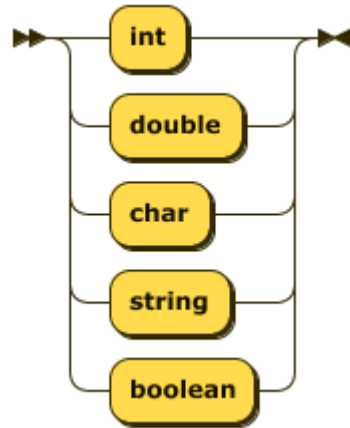
FUNCS



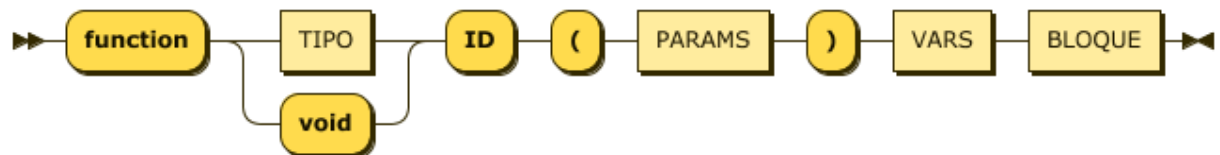
MAIN



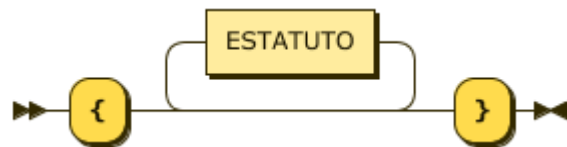
TIPO



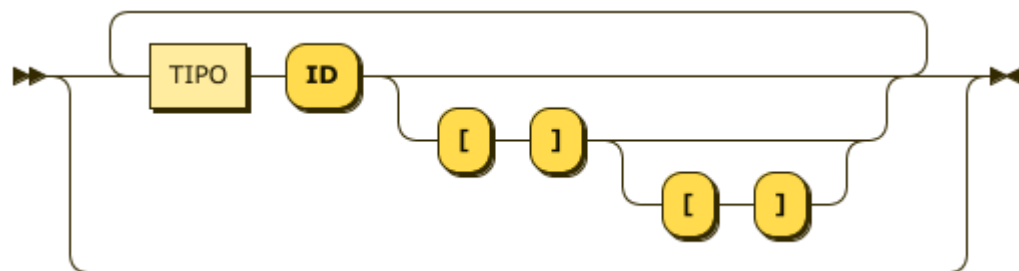
FUNCTION



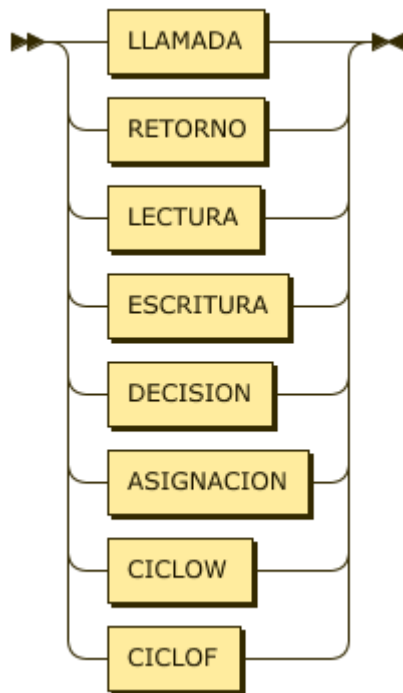
BLOQUE



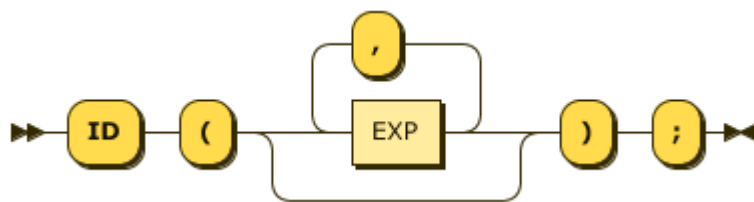
PARAMS



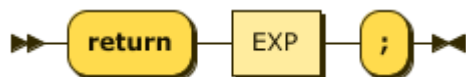
ESTATUTO



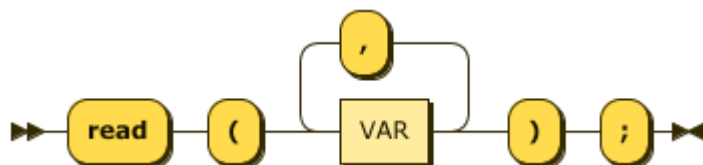
LLAMADA



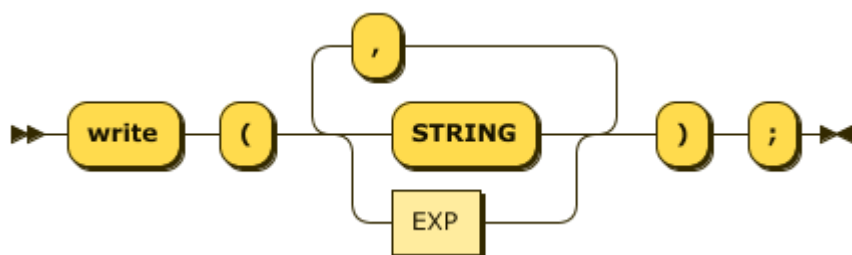
RETORNO



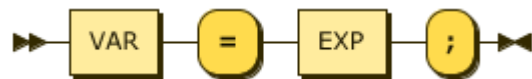
LECTURA



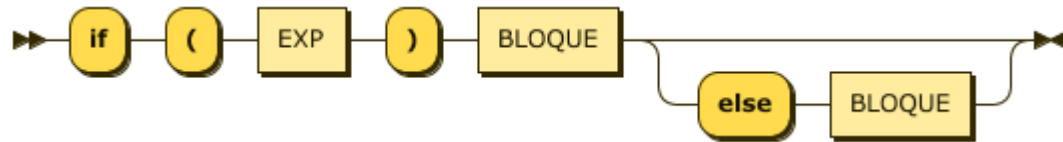
ESCRITURA



ASIGNACION



DECISION



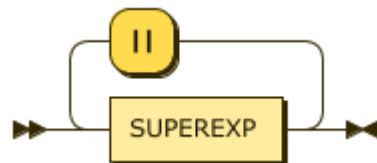
CICLOW



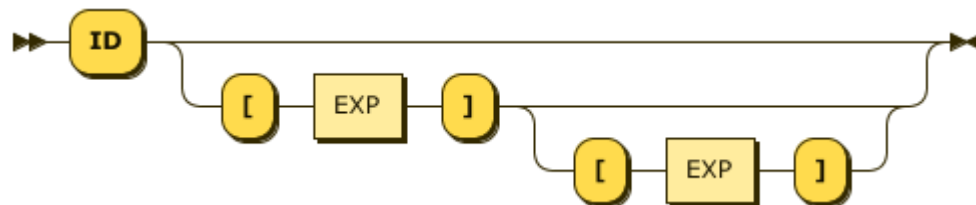
CICLOF



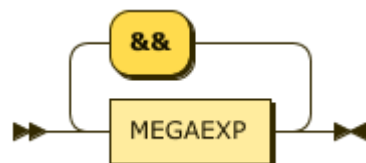
EXP



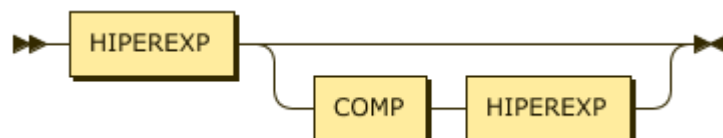
VAR



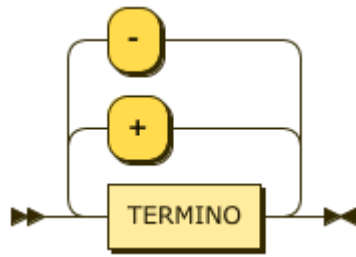
SUPEREXP



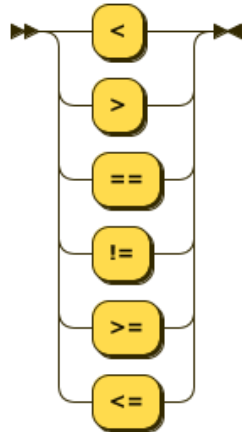
MEGAEXP



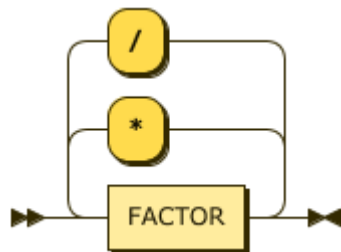
HIPEREXP



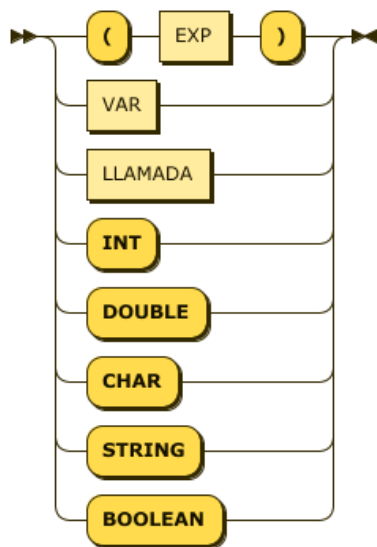
COMP



TERMINO



FACTOR



Terminales

ID

- [a-zA-z]\w*

INT

- [+]? \d+

STRING

- \"[^\"]*\"

DOUBLE

- [+]? \d+ \. \d+

CHAR

- []. []

BOOLEAN

- true | false

Características semánticas

La siguiente tabla representa los tipos que pueden aplicar para cada operador que tenemos en Baky.

Operador	Duplas de tipos válidos
&&	BOOLEAN - BOOLEAN
+ - * /	INT - INT , INT - BOUBLE , DOUBLE - DOUBLE
== !=	INT - INT , INT - BOUBLE , DOUBLE - DOUBLE , STRING - STRING , CHAR - CHAR , BOOLEAN - BOOLEAN
<= >= > <	INT - INT , INT - BOUBLE , DOUBLE - DOUBLE

Funciones especiales

Debido a que Baky es un lenguaje imperativo básico, no cuenta con funciones especiales, sin embargo la complejidad del desarrollo de éstas funciones se pasan al desarrollo del IDE móvil.

Tipos de dato

Como se puede observar en el apartado de nuestros terminales, Baky cuenta con 5 tipos de datos.

- INT
 - Números enteros.
- STRING
 - Secuencia de caracteres.
- DOUBLE
 - Números con decimales.
- CHAR
 - Solo un caracter.
- BOOLEAN
 - Valores booleanos (true | false)

Es importante mencionar que Baky soporta el uso de arreglos.

Lenguaje y OS usado para desarrollo

Baky se desarrollará en el lenguaje de programación Javascript. Esto es debido a que Javascript nos facilitará la integración de Baky en nuestro IDE móvil que de igual manera se desarrollará usando Javascript.

El OS para el desarrollo de Baky será tanto MacOS como Windows debido a los diferentes equipos de cómputo entre los integrantes del proyecto.

Bibliografía

https://www.bottlecaps.de/rr/ui#_Production