

**COMPILADORES 1**

**PROYECTO FINAL**

**PRESENTADO A:**

ING. CARLOS VALLEJO

**ELABORADO POR:**

ALESSANDRO GRIMALDI – 11441226

FRANKLIN GARCIA– 11611001

**ASIGNATURA:**

COMPILADORES I

**LUGAR Y FECHA:** TEGUCIGALPA M.D.C. 9/29/2020

**Introducción**

A través del presente documento presentamos el manual para el proyecto desarrollado en la clase de compiladores 1, teniendo en cuenta muchos aspectos de lenguajes de programacion basicos como bucles, asignaciones y demas fueron implimentados como se requiere según la clase, el proyecto se llama TATO y para este momento esta desarrollado en las 2 primeras fase de un compilador las cuales son lexico y sintactico

**OBJETIVOS**

* Implementar los 2 primeros pasos de un compilador
* Crear un lexema para un lenguaje nuevo
* Implementar la parte sintactica del mismo lexema
* Conocer el concepto de un AST en implementarlo en el proyecto
* Implementar una serie de errores para que el programa detecte los errores posibles cuando se escribe y ejecuta un programa

**TABLA DE DESCRIPCION DEL LEXEMA IMPLEMENTADO Y EJEMPLOS SINTACTICOS PARA EL PROYECTO**

|  |  |  |
| --- | --- | --- |
| ***Token*** | ***Función*** | ***Ejemplo*** |
| Int | Integer | **int** id=4 |
| Un digito seguido de cero o mas digitos | Números | 1,2,3,432 |
| Float | Float | **float** id=> 4.5 |
| String | Cadena | **string** Id=> “alo” |
| Bool | Boolean | **bool** id=> t |
| True | True | bool id=> **true** |
| False | False | bool id=> **false** |
| List | Arreglo de variables | **list** id(int)=>{1,2,3……., n} |
| Matriz | Arreglo bidimensional | **matriz** Id(int)(int) =>{1,2,3…, n}{1,2,3…, n} |
| +, - | OPSUMA | id => 3 **+** 4 |
| / | OPDIV | id => 3 **/** 4 |
| \* | OPMULT | id => 3 **\*** 4 |
| => | OPASIGNACION | id **=>** “a” |
| <,>,=, >=,<=,!=,! | OPRELACION | a **=** b |
| If | Iniciador de bloque de decisión | **If** |
| end if | Fin del if | If condicion then …. **end if** |
| Else | Condición contraria a la dada | **Else** |
| end else | Fin del else | else then …. **end else** |
| else if | Siguiente condición | **else if** condición **then** |
| end else if | Fin del else if | else if condicion then …. **end if else** |
| Then | Make | If condición **then** |
| For | For | **For** |
| In | En | **in** |
| Range | Rango for | **Range** |
| end for | Finalización del for | **end for** |
| While | Loop while | **While** |
| Repeat | Hacer loop while | **Repeat** |
| end while | Fin del while | **end while** |
| ; | PuntoComa | **;** |
| : | DosPuntos | **:** |
| , | Coma | **,** |
| (,) | Parentesis | **()** |
| “,” | Comillas | **“”** |
| return | Retorno | **return** 2; |
| Case | Switch | **case** (condicion) of |
| Of | Casos | case (condicion) **of** |
| default | Default | Default |
| end case | Fin de caso | case (condicion) of …. **end case** |
| Main | Metodo main | **main** begin |
| Begin | Inicio funcion | main **begin** |
| end main | Fin main | main begin … **end main** |
| Println | Imprimir con salto de linea | **Println**(2); |
| Print | Imprimir sin salto de linea | **Print**(2); |
| Call | Llamar metodo | **call** begin ImprimirHola () |
| end call | Fin llamar metodo | callbegin ImprimirHola() …..**end call** |
| Una letra seguida de mas letras o numeros | Identificador | Int **a**=>3; |
| Palabras, numeros o caracteres dentro de comillas | Cadena | String a=> **“hola-+=mundo123”** |
| And, Or | Operadores logicos | If (a > b)And (c>b) |
| Put | Funcion entrada | Put a; |
| /^ | Comentario Entrada | **/^**….^/ |
| ^/ | Comentario Salida | /^….**^/** |

**ESTRUCTURAS IMPLEMENTADAS EN EL PROYECTO**

**Bloque de decisión**

if a=3 then

…..SENTENCIA……

end if

else if a<3 then

…….SENTENCIAS……

else end if

else then

…..SENTENCIAS……

end else

**Bloque de Iteración**

***FOR:***

for Int i = > 0 in range (i < a)

/\*

Loop del FOR

\*/

end for

***WHILE:***

while true repeat

/\*

Loop del WHILE

\*/

end while

**Switch/Case**

case (ID) of

1: Print (“ *Esto se imprimirá si la variable numero es igual a 1.* ”);

2: Print (“ *Esto se imprimirá si la variable número es igual a 20.* ”);

3: Print (“ *Esto se imprimirá si la variable número es igual a 3*. ”);

Default: Print (“ *Esto se imprimirá si no es ninguno de los de arriba* “);

end case

**Objetos**

**Ejemplo (Main-Class-Methods):**

***Class =*** *program*

***Main =*** *main begin*

*end main*

***Método =*** *call*

*end call*

METODO MAIN

main begin

Int numero=>10;

println(numero);

end main

METODO VOID

call begin imprimirHola()

println(“ Hola “);

end call

METODO CON RETORNO

int begin multiplicarNumero(Int numero)

Int multi=>numero\*6;

return numero;

end call

**Método Recursivo**

Int begin metodoRecursivo(Int numero)

numero=>resultado;

if n=1 then

return 1;

end if

else then

resultado => metodoRecursivo(numero-1)\*numero;

return resultado;

end else

end call

**Conclusiones**

* Se conocio y se implemento las primeras fases de un compilador lexico y sintactivo
* Se implemento el AST
* Se creo un lenguaje basado en ideas de otros y nuevas