

DOCUMENTACION PROYECTO PROCESADORES DE LENGUAJE

Juan Jose Corroto

Javier Córdoba

Fernando Vallejo

Beka Bekerí

Hito 1 : 07/10/18

ÍNDICE

1º Lenguaje Mooma P. 3
Tabla de tokens P. 4
Tabla de palabras reservadas P. 4
2º Ejemplo Mooma P. 5
3º Diseño del proyecto P. 6
Anexo I (Reuniones) P. 7

1º Lenguaje Mooma

Nuestro lenguaje para el diseño de maquinas de moore se llama Mooma y viene definido de la siguiente forma:

```
PROGRAM ::= DEFINE,{BLOCK};
BLOCK ::= DEFINE|AUTOMATON|OUTPUT;
DEFINE ::= "define" IDENT "{" ENTRADA ";" SALIDA ";" "}";
AUTOMATON ::= "automaton" IDENT "(" IDENT ")" "{" STATES ";" INITIAL ";" TRANSITIONS "}";
ENTRADA ::= "in" ":" {INPUT};
SALIDA ::= "out" ":" ALFABETO;
STATES ::= "states" ":" [{IDENT "|" INPUT,}|epsilon] IDENT "|" INPUT;
INITIAL ::= "initial" ":" IDENT;
TRANSITIONS ::= "transitions" "{" {IDENT "->" [{IDENT "|" INPUT,}|e] IDENT "|" INPUT ";" } "}";
IDENT ::= LETRA {LETRA|DIGITO};
LETRA ::= "A" | "B" | "C" | "D" | "E" | "F" | "G"
        | "H" | "I" | "J" | "K" | "L" | "M" | "N"
        | "O" | "P" | "Q" | "R" | "S" | "T" | "U"
        | "V" | "W" | "X" | "Y" | "Z" | "a" | "b"
        | "c" | "d" | "e" | "f" | "g" | "h" | "i"
        | "j" | "k" | "l" | "m" | "n" | "o" | "p"
        | "q" | "r" | "s" | "t" | "u" | "v" | "w"
        | "x" | "y" | "z" ;
DIGITO ::= "0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9" ;
INPUT ::= {DIGITO};
OUTPUT ::= "output" ":" CODIGO ";" ;
CODIGO ::= "{" ":" { "." } ":" ;
```

Un dato importante para tener en cuenta es que nuestro lenguaje trabaja con eventos, es decir la creación de diccionario de correspondencia entre eventos y las entradas del problema ha de ser diseñado para la comprensión del resultado.

Los programas Mooma se encuentran divididos en tres partes:

1. Declaración de entradas y salidas (eventos y código)
2. Declaración de las transiciones entre estados, declaración de estados y salida asociada
3. Código asociado a las salidas de los estados.

Aquí dejo la tabla de tokens y la tabla de palabras reservadas correspondiente a nuestro lenguaje Mooma:

Token	Patrón	Lexema de ejemplo
define	define	define
automaton	automaton	automaton
in	in	in
out	out	out
states	states	states
initial	initial	initial
transitions	transitions	transitions
output	output	output
Llave_derecha	}	}
Llave_izquierda	{	{
Punto_y_coma	;	;
Coma	,	,
Asignación	:=	:=
Par_derecho))
Par_izquierdo	((
Flecha	->	->
Código_inicio	{:	{:
Código_final	:}	:}
Separator		
Identificador	[A-Za-z][A-Za-z0-9]*	Foo5
Código	.*	System.out.println("HelloWorld!");

Palabra reservada	Token
define	define
automaton	automaton
in	in
out	out
states	states
initial	initial
transitions	transitions
output	output

2º Ejemplo Mooma

A continuación veamos un ejemplo de definición de una maquina moore en Mooma:

```
define alfabetos1{
    in := 1, 2;
    out := code1, code2;
}

automaton cosal (alfabetos1){
    states := q0|code1, q1|code2;
    initial := q0;
    transitions{
        q0 -> q1|1, q0|2;
        q1 -> q1|1, q0|2;
    }
}

code1 := {
    System.out.println("Hello");
};
code2 := {
    System.out.println("Goodbye");
};
```

Como se puede apreciar solo tenemos dos eventos en este caso (1 y 2), dos estados (q0 y q1) y sus dos salidas junto con el código correspondiente el cual va declarado al final del programa. A parte se puede observar como se declaran las transiciones en nuestro lenguaje, es decir, para cada estado con transiciones se debe declarar de forma que la estructura sea la siguiente

EstadoActual → EstadoX al que llega | si ocurre X evento, ...;

3º Diseño

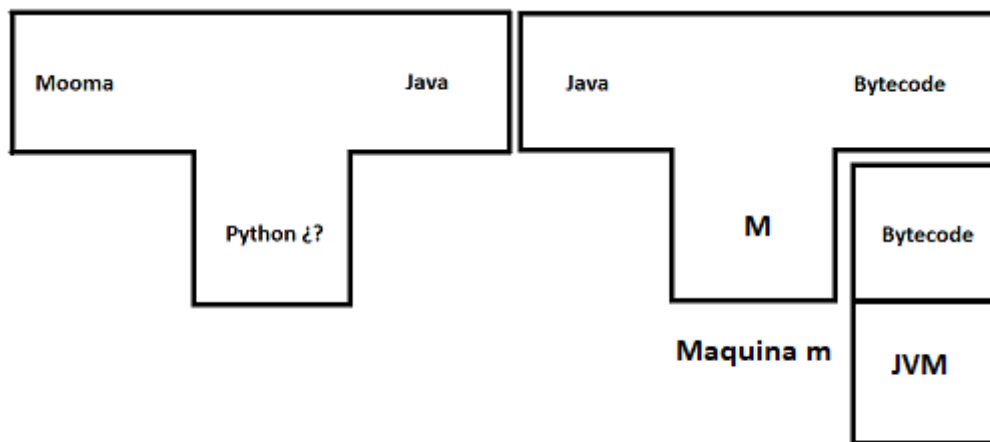
Para nuestro proyecto necesitábamos saber 3 cosas:

- Lenguaje fuente (Diseñado por nosotros). Mooma.
- Lenguaje para el procesador (con el que vamos a trabajar nosotros)
- Lenguaje objeto (al cual vamos a “traducir” nuestro lenguaje)

En cuanto al lenguaje del procesador, de momento creemos que vamos a utilizar Python, debido a que todos lo manejamos y que nos parece un lenguaje útil y versátil a la hora de hacer grandes proyectos, aunque de momento necesitamos concretar los generadores de código que existen para Python.

Y respecto al lenguaje de salida vamos a utilizar Java de forma definitiva.

De modo que nuestro T-Diagrama quedaría de la siguiente forma:



ANEXO I. REUNIONES

Reunión 1

Fecha: 25/09/2018

Hora de inicio: 16:15

Hora de fin: 17: 30

Objetivos de la reunión: Al ser la primera reunión, intentaremos determinar las características de nuestro proyecto (lenguajes a utilizar, maquinas de moore, etc). Por otra parte buscamos la creación de nuestro lenguaje fuente.

Decisiones tomadas: Se terminó el diseño del lenguaje fuente del proyecto. Hemos empezado a buscar lenguajes objeto para el proyecto y hemos barajado algunas posibilidades para el lenguaje de implementación.

Miembros reunidos: Javier Córdoba, Juan Jose Corroto, Fernando Vallejo, Beka Bekerí

Reunion 2

Fecha: 02/10/2018

Hora de inicio: 16:15

Hora de fin: 18:25

Objetivos de la reunión: Una vez que sabemos cual va a ser nuestro lenguaje fuente y que tenemos su diseño completo, podemos pasar a crear el EBNF de nuestro lenguaje. Por otro lado, Juanjo ha traído un ejemplo de maquina de moore en java, para ir viendo la portabilidad de Mooma a Java.

Decisiones tomadas: Java será definitivamente nuestro lenguaje objeto y Python tiene muchas posibilidades de ser nuestro lenguaje de implementación. El EBNF de Mooma esta terminado.

Miembros reunidos: Javier Córdoba, Juan Jose Corroto, Fernando Vallejo, Beka Bekerí