Construcción del lenguaje ñ

LAURA DEL PINO DÍAZ Y AITOR DEL PINO SAAVEDRA HERNÁNDEZ – GRADO EN INGENIERÍA INFORMÁTICA – MENCIÓN EN COMPUTACIÓN

Procesadores de Lenguaje

ÍNDICE

|  |  |
| --- | --- |
| Introducción | 2 |
| Definición del lenguaje Ñ |  |
|  |  |
|  |  |

INTRODUCCIÓN

En este documento definiremos la creación de un nuevo lenguaje de programación denominado Ñ. Este lenguaje estará escrito en español y tendrá una base la cual estará formada por funcionalidades de diferentes lenguajes de programación, entre los que destacan MATLAB, Java y Korn Shell.

Por ello en la siguiente sección modelaremos como va hacer nuestro lenguaje de programación con el objetivo de poder crear nuestro compilador en C.

DEFINICIÓN DEL LENGUAJE Ñ

TIPOS DE DATOS

En esta sección definiremos el modelo de nuestro lenguaje. El lenguaje Ñ estará formado por tres tipos de datos:

* Reales. Este tipo lo utilizaremos para codificar cualquier número.
* Caracteres. Este tipo nos va a permitir poder representar lo caracteres del código ASCII.
* Vectores. Este tipo lo utilizaremos para almacenar un conjunto de caracteres o de números en un espacio de memoria contigua y que podremos referenciar mediante un nombre.

|  |  |
| --- | --- |
| Tipos. | Ejemplos. |
| Reales. | 5, -1 , 3.05 |
| Caracteres. | A,b,5,-,\* |
| Vectores. | [1,-2,3.05,-50.10],[H,o,l,a] |

Para declarar una variable de tipo real se puede utilizar las siguientes gramáticas:

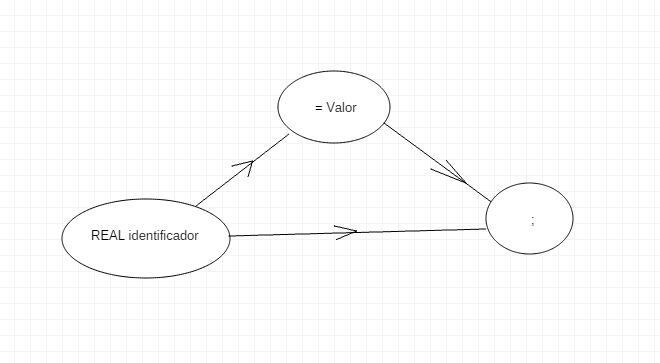


Ilustración : declaración de una variable real.

<falta la declaración de caracteres y vectores, que supongo que serán como Java y MatLab respectivamente>

OPERADORES ARITMÉTICOS

En este lenguaje dispondremos de un conjunto de operadores bastante grandes, lo que nos permitirá realizar prácticamente cualquier operación de manera directa. Dispondremos de operadores aritméticos, relacionales y lógicos.

Los operadores aritméticos los utilizaremos para realizar operaciones de tipo numérico por tanto se utilizarán el tipo real o un vector con un conjunto de números.

|  |  |  |
| --- | --- | --- |
| Operador | Descripción | Ejemplo |
| + | Suma | 5 + 2 = 7 |
| .+ | Suma punto | [1,2] + [2,3] = [3,5] |
| - | Resta | 5-3 = 2 |
| .- | Resta punto | [1,3.5,2] – [1,0,-3] = [0,3.5,-1] |
| \* | Producto | 3 \* 5 = 15 |
| .\* | Producto punto | [1,2,5]\*[1,2,3] = [1,4,15] |
| / | División | 20/5 = 4 |
| ./ | División punto | [10,15,30] ./ [2,3,10] = [5,5,3] |
| % | Módulo | 4%2 = 0 |
| .% | Modulo punto | [4,3,1] .% [4,3,1] = [0,0,0] |
| ++ | Incremento | 5++ = 6 |
| -- | Decremento | 5-- = 4 |

Además podremos realizar sumas, restas, multiplicaciones y divisiones de un valor por un vector, de manera que el resultado será realizar la operación de ese valor por todos los valores que componen el vector. Por ejemplo 5 + [1,2,3] = [6,7,8].

OPERADORES LÓGICOS

También tendremos operadores relaciones los cuales utilizaremos para realizar una comparación numérico o de caracteres entre dos operandos. Exactamente los caracteres utilizarán únicamente los operadores de igual o distinto. El resultado de la evaluación de cualquier operador relacional será 1 o 0, 1 si se cumple esa comparación y 0 sino se cumple.

|  |  |  |
| --- | --- | --- |
| Operador | Descripción | Ejemplo/Resultado |
| == | Igual | 5 == 5 /1 |
| ¡= | Distinto | a != b /0 |
| > | Mayor | 5 > 0 /1 |
| < | Menor | 5 < 4 /0 |
| >= | Mayor o igual | 5 >= 2 / 1 |
| <= | Menor o igual | 10 <= 5 / 0 |

OPERACIONES SOBRE ARREGLOS

Del lenguaje MatLab hemos decidido adoptar, por la claridad del código, la siguientes sintáxis :