

ManualLenguajeJmery++

Integrantes: Juan Pablo Martínez

Michael Bravo Mery

Profesor: Eric Ross

Asignatura: Lenguajesde programación

Introducción general:

El lenguaje Jmery++ es un lenguaje de programación básico el cual puede facilitar el cálculo de las cuatro operaciones básicas de la matemática:

1-sumar

2-restar

3-multiplicacion

4-dividir

Funcionamiento:

La forma de poder compilar un programa en Jmery++ y ejecutarlo se muestra en un video en la siguiente URL:

<https://www.youtube.com/watch?v=L6CJG9Ji2OU&feature=youtu.be>

1-Sintaxis Jmery:

Lo primero para empezar a programar en este lenguaje se inicializa con la palabra INICIO_JMERY y se termina con la FIN_JMERY como en cualquier lenguaje común ejemplo:

INICIO_JMERY

//declaración de variables

{

// instrucciones

}FIN_JMERY

1.1 Declaración de variables

Este lenguaje solo soporta variables de tipos enteras y String las cuales se declaran de la siguiente manera:

Todas las declaraciones de van después de la palabra INICIO_JMERY luego de declarar las variables se abre una llave (NOTA: antes de la llave solo se declaran las variables sin valores los valores se le asignan después de la llave).

Sintaxis de declaración de variable

INTJMERY nombreVariable;

STRINGJMERY nombreVariable;



//EJEMPLO

INICIO_JMERY

INTJMERY contador;

STRINGJMERY x;

{contador = 0;

 x = "hola";

} FIN_JMERY

Por otro lado, este lenguaje tiene la posibilidad de realizar ciclos en el que la sintaxis es la siguiente:

```
Palabra reservada ITERA<<condición>>{  
                                     //instrucción  
}
```

La asignación de valores a una variable es de la siguiente manera:

Variable1=variable2;➡ahora la variable1 tiene el valor de la variable2 .

Las operaciones básicas tienen la siguiente sintaxis:

Suma = variable1 + variable2;

En el caso de la resta, multiplicación y división sería lo mismo solo cambiando el símbolo del operador por resta (-), multiplicación (*) y división (/)

También está la sentencia condicional if tiene la siguiente sintaxis:

```
JMERY_IF<<condición>>{      // instrucción }
```

Ejemplo:

```
JMERY_IF<<1 == 1>>{  
    //INSTRUCCION  
}
```



Finalmente, las impresiones tienen la siguiente sintaxis

```
imprimeInt<< variableTipoEntera >>
```

```
imprimeString<< variableTipoString o "texto">>
```

Nota las sentencias de impresiones son las únicas que van sin el punto y coma al final