



**UNIVERSIDAD NACIONAL EXPERIMENTAL
DE LOS LLANOS OCCIDENTALES
“EZEQUIEL ZAMORA”
UNELLEZ-BARINAS**

MODULO IV PROGRAMACIÓN FUNCIONAL Y LÓGICA

PROFESOR:

ING. RONALD GIL

SUB-PROYECTO:

DESARROLLO DE APLICACIÓN I

BACHILLERES:

GIANNA MOFFA

C.I: 26.103.484

RANGEL LEONARDO

C.I: 22.984.725

ING. EN INFORMÁTICA

BARINAS, MARZO DEL 2018

PROGRAMACIÓN FUNCIONAL

Es un paradigma de programación declarativa basado en el uso de funciones matemáticas, en contraste con la programación imperativa, que enfatiza los cambios de estado mediante la mutación de variables. Los lenguajes de programación funcional, especialmente los puramente funcionales, han sido enfatizados en el ambiente académico y no tanto en el desarrollo comercial o industrial. Sin embargo, lenguajes de programación funcional como Scheme, Erlang, Rust, Objective Caml, Scala, F# y Haskell, han sido utilizados en aplicaciones comerciales e industriales por muchas organizaciones.

La programación funcional también es utilizada en la industria a través de lenguajes de dominio específico como R (estadística), Mathematica (matemáticas simbólicas), J y K (análisis financiero). Los lenguajes de uso específico usados comúnmente como SQL y Lex/Yacc, utilizan algunos elementos de programación funcional, especialmente al procesar valores mutables. Las hojas de cálculo también pueden ser consideradas lenguajes de programación funcional.

CARACTERÍSTICAS ESENCIALES

Los programas escritos en un lenguaje funcional están constituidos únicamente por definiciones de funciones, entendiendo éstas no como subprogramas clásicos de un lenguaje imperativo, sino como funciones puramente matemáticas, en las que se verifican ciertas propiedades como la transparencia referencial (el significado de una expresión depende únicamente del significado de sus expresiones), y por tanto, la carencia total de efectos colaterales.

Otras características propias de estos lenguajes son la no existencia de asignaciones de variables y la falta de construcciones estructuradas como la secuencia o la iteración (lo que obliga en la práctica a que todas las

repeticiones de instrucciones se lleven a cabo por medio de funciones recursivas).

Existen dos grandes categorías de lenguajes funcionales: los funcionales puros y los híbridos. La diferencia entre ambos estriba en que los lenguajes funcionales híbridos son menos dogmáticos que los puros, al admitir conceptos tomados de los lenguajes imperativos, como las secuencias de instrucciones o la asignación de variables. En contraste, los lenguajes funcionales puros tienen una mayor potencia expresiva, conservando a la vez su transparencia referencial, algo que no se cumple siempre con un lenguaje funcional híbrido.

Ejemplo:

```
1  <?php
2      $operacion = $_POST['operacion'];
3      $numero1 = $_POST['numero1'];
4      $numero2 = $_POST['numero2'];
5      $result;
6      $verificar = array($numero1);
7      foreach($verificar as $datos){
8          if (is_numeric($datos))
9          {
10             if ($operacion=="suma")
11             {
12                 $result= $numero1+$numero2;
13             }
14             else if($operacion=="restar")
15             {
16                 $result= $numero1-$numero2;
17             }
18             else if($operacion=="multiplicacion")
19             {
20                 $result= $numero1*$numero2;
21             }
22             else
23             {
24                 $result= $numero1/$numero2;
25             };
26         }
27         else
28         {
29             echo "SOLO NUMEROS";
30         }
31     }
32     echo $result;
33     ?>
```

PROGRAMACIÓN LÓGICA

La programación lógica es un tipo de paradigmas de programación dentro del paradigma de programación declarativa. El resto de los subparadigmas de programación dentro de la programación declarativa son: programación funcional, programación con restricciones, programas DSL (de dominio específico) e híbridos. La programación funcional se basa en el concepto de función (que no es más que una evolución de los predicados), de corte más matemático. La programación lógica gira en torno al concepto de predicado, o relación entre elementos.

CARACTERÍSTICAS ESENCIALES

- 1- Comprende la programación declarativa y la programación funcional.
- 2- El programador se concentra más en el conocimiento que en los algoritmos.
- 3- Se basa en el concepto de función (que no es más que una evolución de los predicados), de corte más matemático.
- 4- La programación lógica gira en torno al concepto de predicado, o relación entre elementos.

Ejemplo:

```
1  function member(item:Integer; L:Array of Integer):Boolean;  
2      var  
3          i:Integer;  
4      begin  
5          i:=0;  
6          while((i <= length(L)) and (L[i] <> item))  
7              do  
8                  begin  
9                      i := i+1;  
10                 end;  
11                 Result := item = L[i];  
12             end;
```