**Universidad Tecnológica de México**

**Verduzco López Rubén**

**Ing. Sistemas Computacionales**

**Programación Lógica**

**Jesús Barajas Natividad**

**Tarea 3.- Recursividad**

****

**Recursividad**

El concepto de recursividad es un concepto muy abstracto y complejo que tiene que ver tanto con la lógica como también con la matemática y otras ciencias. Podemos definir a la recursividad como un método de definir un proceso a través del uso de premisas que no dan más información que el método en sí mismo o que utilizan los mismos términos que ya aparecen en su nombre, por ejemplo, cuando se dice que la definición de algo es ese algo mismo.

La recursividad tiene como característica principal la sensación de infinito, de algo que es continuo y que por tanto no puede ser delimitado en el espacio o el tiempo porque se sigue replicando y multiplicando de manera lógica y matemática. Así, es común encontrar casos de recursividad por ejemplo en imágenes de espejos que hacen que la imagen sea replicada al infinito, una dentro de otra hasta que deja de verse, pero no por eso deja de existir.

Ejemplos:

*En Prolog con un ejemplo muy conocido en la recursividad, como lo es obtener el factorial de un número.*

*Factorial*

*factorial(3,6).*

*factorial(Num,Final) :-*

*Num>0,*

*Num1 is Num-1,*

*factorial(Num1,Final1),*

*Final is Num \* Final1.*

*La Sucesión de Fibonacci es una sucesión infinita de números, donde cada termino este dado por la suma de los dos anteriores, empezando en 1,1, y luego la sumas.,*

*1,1,2,3,5,8,13,21, 34...*

***Fibonacci***

*fibonacci(0,0).*

*fibonacci(1,1).*

*fibonacci(N,Y):-N>1,*

*N1 is N–1,*

*fibonacci(N1,Y1),*

*N2 is N–2,*

*fibonacci(N2,Y2),*

*Y is Y1+Y2.*

**Conclusiones:**

Aplicando la recursividad se obtienen mejores resultados, porque a la hora de aplicar este concepto se reduce la cantidad de código cuando se programa, también se puede ocupar ciclos con la recursividad, se debe tener cuidado porque se puede generar un bucle y descontrol en el programa.

**Referencias:**

<https://es.wikipedia.org/wiki/Recursi%C3%B3n>

<http://smartcomputing.gerardorossel.org/reglas-recursivas.aspx>

<https://proauge.com/2015/08/30/recursividad-en-prolog/>