



## INGENIERÍA EN SISTEMAS COMPUTACIONALES

3er Semestre

20/09/2019

Actividad 1: Mapa Conceptual.

Tema 2: Recursividad.

Materia: Estructura de Datos.

Nombre del Alumno: Marín Ramírez Mario.

Número de Control: S18070186

Correo electrónico: mariomarin502t@gmail.com

Profesor: I.S.C Acevedo Sandoval Salvador

## **CUESTIONARIO**

1. ¿Qué es la RECURSIVIDAD?

Propiedad que posee un método por la cual puede llamarse a sí mismo.

¿Para qué se utiliza la recursividad?

Como una alternativa a la iteración, una solución recursiva es, normalmente, menos eficiente en términos de tiempo de computadora que una solución iterativa.

¿Qué ventajas y desventajas existen al utilizar recursividad?

Ventaja: permite a los programadores especificar soluciones naturales, sencillas, que serían, en caso contrario, difíciles de resolver

Desventaja: menos eficiente en términos de tiempo de computadora que una solución iterativa, debido a las operaciones auxiliares que llevan consigo las invocaciones suplementarias a los métodos

4. ¿Qué es un procedimiento/función recursivo?

Es aquel que se llama a sí mismo, bien directamente o bien indirectamente, a través de otro método.

5. ¿Cuáles son las partes que conforman un funcion/procedimiento recursivo?

Caso Base.

Progreso.

6. ¿Por qué son importantes estas partes que lo conforman?

Porque existe alguna solución para el problema integrado y cualquier llamado a sí mismo puede ser ejecutado.

7. ¿Cuáles son los pasos para implementar un procedimiento recurso?

Implementación de la clase/método.

Creación del algoritmo.

Aplicar operaciones con el método invocado.

8. Tipos de recursividad

De Cabeza

De cola

Intermedia

Múltiple

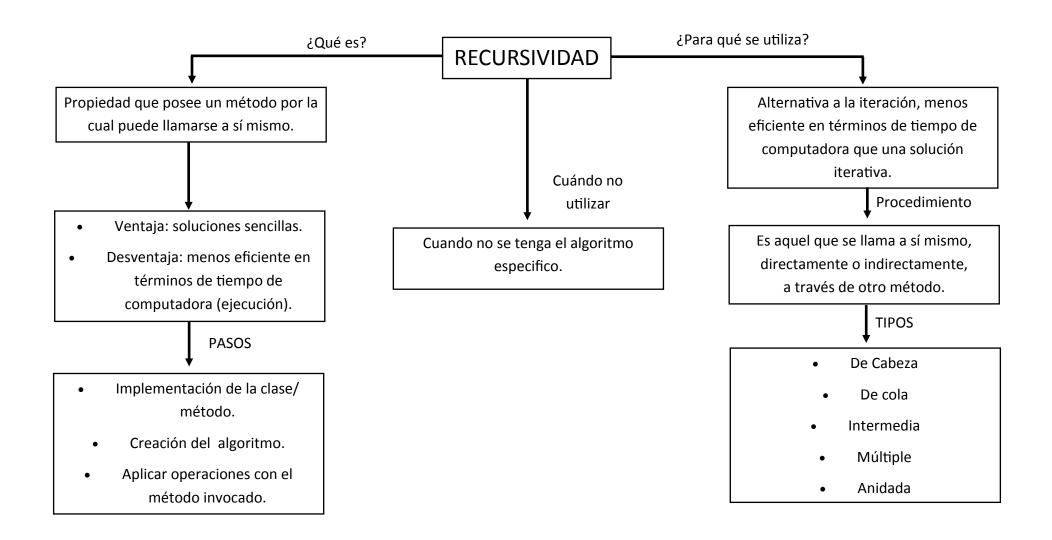
## Anidada

9. ¿Qué consideraciones tomaría en cuenta para resolver un problema mediante un algoritmo recursivo?

La inicialización de las variables y del tiempo de ejecución, necesariamente saber el tipo de problema y el algoritmo que sea.

10. ¿Definitivamente cuándo no aplicaría un algoritmo recursivo?

Cuando no se tenga el algoritmo específicamente pensado, se hace más complicado el minimizar código y el tamaño de ejecución.



Referencia:
Luis Joyanes Aguilar, Ignacio Zahonero Martínez. (2000). Tipos de Datos. En Estructura de Datos en Java(561). España.: S.A. MCGRAW-HILL / INTERAMERICANA DE ESPAÑA.