Aplicación De Métodos de Ordenamiento En Grandes Cantidades De Datos Aplicado A La Programación Orientada A Objetos

Heriberto David Yepes – herdavyep@hotmail.com

Cristhian Urrego Salazar – cristhianu92@gmail.com

**Resumen: aplicando los cuatro pilares de la programación orientada a objetos utilizando Java como lenguaje de programación, se realizo la construcción de un software que implementa la creación de archivos con grandes cantidades de números enteros de cinco cifras al azar utilizando el método *Random* para posteriormente ordenarlos con diferentes métodos de ordenamiento con el objetivo de realizar un análisis y determinar cual es el método mas eficiente de los que se implementaron en el proyecto**

Palabras claves: Java, objetos, ordenamiento, orientada, métodos, P.O.O, programación.

Abstract

Introducción

En el presente trabajo se desarrolla una implementación de distintos métodos de ordenamiento con el objetivo de entender mejor su función y eficiencia en grandes cantidades de números enteros de hasta veinte millones de datos realizando una comparativa entre ellos concluyendo cual de ellos es el mas eficiente. Dividiendo el proyecto en cinco clases principales y una interface padre que hereda a las clases de los métodos de ordenamiento.

Clase menú, Interface principal de los métodos de ordenamiento, Conversión de tiempo, Clase de métodos de archivos de texto.

I. Metodología Utilizada

Los algoritmos de ordenamiento nos permite, como su nombre lo dice, ordenar. En este caso, nos servirán para ordenar vectores o matrices con valores asignados aleatoriamente. Nos centraremos en los métodos más populares, analizando la cantidad de comparaciones que suceden, el tiempo que demora y revisando el código, escrito en Java, de cada algoritmo.

II. Estructura del Programa

En el desarrollo del software se implemento como clase principal el *RunMain* el cual se encarga de ejecutar el método estático *Menu.getMenuPrincipal* que desencadena todo el funcionamiento del programa. La siguiente de las clases a tomar en cuenta en un correcto orden de desarrollo es la clase *Menu* que se ocupa de imprimir los formularios de opciones dentro del programa así como de llamar a los métodos de las demás clases, aquí se crea el archivo de fichero de texto llamado *Lista.txt* para llenarlo y trasladar la información del archivo a un *array* por la clase  *MetodosArchivoTxT* para su posterior ordenamiento. Después tenemos la *InterfaceMetodo* que hereda a todas las clases de algoritmos de ordenamiento he implementa la clase *ConversionTiempo* que recibe una variable con la cual ejecuta un mensaje al usuario con el tiempo que tardo la ejecución del método de ordenamiento.

Algoritmos utilizados $$

Lenguaje de programación utilizado

Código de los algoritmos utilizados $$

Resultados alcanzados

Evaluación de los resultados de los algoritmos implementados

Conclusiones

Recomendaciones

Computador usado $$

Referencias bibliográficas