



INSTITUTO POLITÉCNICO NACIONAL



ESCUELA SUPERIOR DE CÓMPUTO

Algoritmos y Estructuras de Datos

"Primera evaluación práctica"

"Algoritmos de ordenamiento"

Alumnos: Rivera Ramos Roberto Iván

Grupo: 2CV3

Profesora: Sánchez García Octavio

Link de GitHub: https://github.com/S3R4PH1N3/1ra_Evaluaci-n.git

Algoritmo de ordenamiento

Un algoritmo de ordenamiento es un algoritmo que pone los elementos de una lista o vector en una secuencia dada, por una relación de orden, es decir, muestra el resultado reordenando la lista o vector.

Algoritmo burbuja / Bubble sort

El algoritmo de ordenamiento de burbuja es uno de los algoritmos de ordenamiento más sencillos aunque no el más eficiente.

Ordenamiento por selección / Selection sort

Este algoritmo mejora ligeramente el algoritmo de burbuja.

Ordenamiento por inserción / Insertion sort

El ordenamiento por inserción, cada elemento nuevo se inserta de vuelta en la sublista ordenada, el algoritmo es muy eficiente para la clasificación de números pequeños de elementos.

Algoritmos de ordenamiento

#	Burbuja	Selección	Inserción
1	0.07400 secs	0.08500 secs	0.095000 secs
2	0.07200 secs	0.08000 secs	0.09600 secs
3	0.09200 secs	0.20200 secs	0.21200 secs
4	0.09100 secs	0.09800 secs	0.20800 secs
5	0.07700 secs	0.08200 secs	0.09300 secs
6	0.21600 secs	0.09200 secs	0.20500 secs
7	0.09000 secs	0.20000 secs	0.206000 secs
8	0.09400 secs	0.20200 secs	0.092000 secs
9	0.08800 secs	0.09200 secs	0.20300 secs
10	0.09400 secs	0.20200 secs	0.22500 secs
11	0.07300 secs	0.08200 secs	0.08900 secs
12	0.09400 secs	0.20000 secs	0.09400 secs
13	0.20700 secs	0.22500 secs	0.21000 secs
14	0.09600 secs	0.20200 secs	0.09600 secs
15	0.08800 secs	0.09900 secs	0.09300 secs

Si sumamos todos los valores obtenidos, tenemos

Ordenamiento burbuja: 2.346 segundos

Ordenamiento por selección: 1.418 segundos

Ordenamiento por inserción: 1.507 segundos

Entonces podemos decir que el algoritmo de ordenamiento de burbuja es el que menos tarda en ordenar la lista.

Ordenamiento burbuja

Es uno de los algoritmos mas simples que se conocen. Se basa en una serie de intercambios entre elementos adyacentes. Estos intercambios dan la impresión de que cada elemento va ascendiendo a través del array acercándose cada vez mas a su posición final.