



Algorítmica y Programación I

Trabajo Práctico N°6 – Ordenamiento y Búsqueda

Ejercicio 1

Desarrolle el procedimiento para el método de ordenamiento que a continuación se describe:

- Que encuentre el máximo elemento del arreglo y lo intercambie con el último.
- Que encuentre el máximo entre los N-1 restantes y lo intercambie con el penúltimo.
- Que continúe así hasta obtener el vector ordenado.

Aplique el procedimiento en un programa que ordene una lista de nombres.

Ejercicio 2

En un arreglo se encuentran cargados en forma desordenada los niveles de audiencia (representados como números enteros) de cierto programa televisivo. Estos niveles no se repiten. Programe el algoritmo BURBUJA para emitir un listado de estos niveles ordenados en forma DECRECIENTE. Indique además si algún nivel de audiencia es igual a uno ingresado por teclado. Para esto último utilice Búsqueda Binaria.

Ejercicio 3

Codifique el algoritmo Burbuja Bidireccional para ordenar un arreglo de cadenas de caracteres. Ordene el listado de los alumnos de la clase e imprima el vector original (desordenado) y el vector ordenado.

Ejercicio 4

Desarrolle el programa en pseudocódigo que ingrese los códigos numéricos con que se individualizan en la Biblioteca de la sede los libros, y:

- Ordene la información en forma CRECIENTE, guardando su posición original en otro arreglo auxiliar.
- Muestre la información ordenada, junto con la posición original de cada ejemplar.
- Busque la posición original de cualquier libro solicitado por teclado e imprímala.

Ejemplo:

| | | | | | | | | | | | | | | |
|-----------------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Información Original: | 89 | 17 | 86 | 75 | 45 | 34 | 65 | 78 | 43 | 99 | 87 | 15 | 21 | 33 |
| Información Ordenada: | 15 | 17 | 21 | 33 | 34 | 43 | 45 | 65 | 75 | 78 | 86 | 87 | 89 | 99 |
| Posición Original: | 12 | 2 | 13 | 14 | 6 | 9 | 5 | 7 | 4 | 8 | 3 | 11 | 1 | 10 |

¿Cuál es la posición original del 43? Es 9. ¿Y del 89? Es 1.

Ejercicio 5

Se dispone de una matriz que contiene en una columna las estaturas de los alumnos de una clase. Se desea calcular la media, y agregar a cada altura un valor adicional (en otra columna) que indica su relación con la media: -1 si la altura es menor que la media, 0 si la altura es igual a la media, y 1 si la supera. Además, se desea imprimir la matriz en forma ordenada, creciente, por altura. La matriz original está desordenada.

Ejercicio 6

Una fábrica de materiales eléctricos posee X códigos de sus productos asentados en planillas de cálculo. Debido a la incorporación de nuevas tecnologías, fabrica otros Y items que coloca a continuación de los primeros. Los códigos son de 3 dígitos. Se decidió sistematizar el control de stock, para lo cual se necesita un programa que:

- a. Ingrese la cantidad de items de cada grupo (las cantidades X e Y).
- b. Lea el vector formado por los Z ($Z=X+Y$) códigos. Esta información está ordenada por grupo, pero desordenada en cuanto a numeración del código.
- c. Genere dos vectores con el vector anterior, tales que uno contenga los códigos viejos y el otro, los códigos de los productos nuevos.
- d. Ordene en forma decreciente los vectores del punto anterior.



-
- e. Sabiendo que los códigos que comienzan con un 5 representan “cables de alta tensión”, programar el algoritmo que permita listar los códigos correspondientes a todas las variedades en stock de dichos cables.

Ejercicio 7

Programar el algoritmo que sea capaz de medir la eficiencia de los métodos de búsqueda secuencial y binaria, al buscar un elemento dado dentro del mismo vector por ambos métodos. Tener en cuenta las siguientes condiciones:

- Las componentes del vector deben generarse en forma aleatoria con valores enteros.
- Debe realizar búsquedas de elementos situados entre los primeros, entre los centrales y los últimos, a fin de obtener una media de las comparaciones.
- La eficiencia estará dada por el número de comparaciones realizadas para encontrar un elemento.
- ¿Qué pasa con la eficiencia cuando el elemento que buscamos no está?

Ejercicio 8

Escriba un algoritmo que lea una sucesión de valores numéricos, que ingresará el operador del programa, y ordene por INSERCIÓN dicho vector.

Imprima el vector de datos de entrada y el resultante de su ordenamiento.

Ejercicio 9

Implementar el algoritmo de Ordenamiento BucketSort, y el programa que permita ejecutarlo. Genere, además, un conjunto de datos para probarlo, que supere los 200 elementos.

Ejercicio 10

Escriba un programa que pida por teclado una matriz de letras, de $N \times M$, y luego ordene cada fila en orden ascendente (usando el método que prefiera). Imprima la matriz recibida y la matriz ordenada