

FLIP SORT

Sara Chica, Rodrigo Gualtero

01 de Diciembre, 2012

Índice

1. Introducción	1
2. Definición del problema	2
2.1. Entrada	2
2.2. Salida	2
3. Modelamiento matemático	2
4. Planteamiento de la Solución	3
5. Conclusiones	3

1. Introducción

Este es un problema de la UVA, identificado con el código *10327*, en el cual se desea saber la cantidad de pasos mínimo para ordenar un conjunto de números enteros en forma ascendente.

A continuación se presentan 2 ejemplos de nodos:

En el primer ejemplo se presenta un conjunto ordenado de 3 números:

$$A = \{1, 2, 3\}$$

Ejemplo 1

Por lo tanto no se necesitan pasos para organizarlo.

En el segundo ejemplo se presenta un conjunto desordenado de 3 números:

$$B = \{2, 3, 1\}$$

Ejemplo 2

Se necesitan 2 pasos para ordenarlo.

2. Definición del problema

Este problema busca ordenar un conjunto de números en forma ascendente, es decir de menor a mayor.

2.1. Entrada

Entra cada uno de los casos de prueba en dos partes. La primera consta de un número que indica la cantidad de enteros que se deben organizar, y finalmente entran todos números en el orden actual.

2.2. Salida

Imprime la siguiente frase:

Minimum exchange operations : X.

donde X es un entero que indica la cantidad de pasos que tuvo que hacer para organizar el vector.

3. Modelamiento matemático

Se sabe que el tren está organizado si:

$$\text{Ordenado} \equiv (\forall i, j | 0 < i < j < 100 : V_{i+1} = V_j)$$

Sobre este problema se podría extender un árbol de posibilidades, el cual tendría todas las posibles permutaciones que habrían, donde una de estas es la que satisface la condición de salida: Tener todos los números ordenados. Aplicando esto ambos ejemplos se tendría el siguiente árbol de posibilidades:



Árbol de posibilidades para un conjunto de 3 enteros.

4. Planteamiento de la Solución

Aunque abrir un árbol de posibilidades permite analizar mejor el problema, no es la mejor solución, debido a que es poco eficiente; es por ello que existen varios algoritmos de ordenamiento, los cuales permiten organizar una lista en una secuencia dada. Para este problema se sugiere algún algoritmo de Intercambio, por lo cual se usará el algoritmo burbuja.

Ordenamiento por Burbuja: Consiste en organizar el vector revisando cada elemento de la lista y comparándolo con el siguiente, de esta forma si están en el orden equivocado se deben intercambiar.

Para el ejemplo 2 se tendrían los siguientes pasos

Origina: $A = \{2, 3, 1\}$

Paso 1: $A = \{2, 1, 3\}$

Paso 2: $A = \{1, 2, 3\}$

5. Conclusiones

1. Conocer los algoritmos de ordenamiento es importante para optimizar algunos otros algoritmos como los de búsqueda y fusión; ya que estos pueden requerir listas ordenadas para tener una ejecución rápida.
2. El método de ordenamiento burbuja es el algoritmo de ordenamiento más usado en los lenguajes de programación.