



aprenderaprogramar.com

# Ejercicio de algoritmia resuelto con Visual Basic: ordenar series de series de números. (CU00356A)

Sección: Cursos

Categoría: Curso Visual Basic Nivel I

Fecha revisión: 2029

Autor: Mario R. Rancel

Resumen: Entrega nº55 del Curso Visual Basic Nivel I

29

## Ejercicio complementario y opcional para el curso Programación en Visual Basic. Nivel I.

### EJERCICIO DE ALGORITMIA RESUELTO CON VISUAL BASIC: ORDENAR SERIES DE SERIES DE NÚMEROS

El ejercicio consiste en generar un programa enfocado al programador (un programa para verificar el funcionamiento del algoritmo y no para realizar un proceso orientado al usuario) capaz de ordenar una serie de series de números de mayor a menor.

El objetivo del programa es realizar un proceso análogo al de ordenar palabras por orden alfabético, pero en este caso usando números. Consideraremos que una serie de números es mayor que otra cuando un elemento de esa serie en igual posición que otros es mayor que ellos dentro del orden de prelación definido por las columnas. En caso de empate entre dos series, el desempate se dirime evaluando el siguiente elemento de la serie sólo entre las series empatadas.

**Ejemplo 1:** Serie 1: 100, 45, 33, 2500  
Serie 2: 111, 22, 112, 335  
Serie 3: 44, 22, 1, 3

Las series se ordenarían así: serie 2 > serie 1 > serie 3, en base a que tras comparar los elementos nº1 de cada serie, ha resultado ese orden y no existen empates.

**Ejemplo 2:** Serie 1: 5500, 35, 23, 6500  
Serie 2: 5500, 22, 11, 3345  
Serie 3: 43500, 33, 2, 2

Las series se ordenarían así: serie 3 > serie 1 > serie 2, en base a que tras comparar los elementos nº1 resulta que es la serie 3 la primera y queda como empate la segunda y tercera series. Al existir empate en un elemento, se pasa al siguiente pero sólo entre aquellas series que han resultado empatadas. Del análisis del elemento nº2 resulta que es mayor en la serie 1 que en la serie 2, con lo cual la serie 1 pasa a ocupar el orden segundo y la serie 2 el orden tercero.

### SOLUCIÓN

El trasfondo de este programa gira en torno a la siguiente pregunta: ¿Cómo podríamos ordenar palabras si el lenguaje informático con el que trabajamos no fuera capaz de comparar dos palabras? Si empiezas a manejar cadenas (palabras) con Visual Basic comprobarás que ordenar palabras es relativamente sencillo gracias a que el lenguaje es capaz de comparar dos palabras y decir cuál es "mayor" y cuál es "menor" (orden alfabético). Si el lenguaje no fuera capaz de esto, probablemente recurriríamos a tratar de transformar las letras en números, quedándonos con un problema de ordenar números de solución aparentemente trivial.

Hasta ahí parece todo claro. La variante que se nos presenta es que los problemas de ordenar números que hemos tratado hasta ahora consisten en **ordenar una serie** de números mientras que ahora queremos **ordenar series** de números. La dificultad radica en que es necesario ir comparando más o menos elementos y series en función de la existencia o no de empates.

La primera cuestión a valorar es que el problema goza de cierta complejidad y por tanto, como siempre pero con más razón si cabe, debemos dedicar tiempo y esfuerzo a su planteamiento y resolución sobre el papel antes de ponernos a programar. En caso contrario (puedes probarlo), con mucha probabilidad nos veremos atrapados en un maremagnum de código con el que creemos estar cerca de la solución a la que sin embargo no llegamos. Y así pasan horas... y días... y si te descuidas semanas.

Una cuestión a valorar dentro de los aspectos previos es: **¿El problema se puede resolver a través de un algoritmo?**

Como paso preliminarísimo nos podemos preguntar si es posible definir un algoritmo para resolver el problema. Para responder a esto, pensemos en cómo ordenamos palabras los humanos... Cojamos un lápiz y un papel y escribamos una lista de palabras. Procedamos a ordenarlas, marcando junto a cada una de ellas el orden que ocupan en la lista (1, 2, 3...) ¿Estaremos siguiendo un proceso lógico y ordenado? La respuesta es sí, con lo cual "podemos programarlo". ¿Seguimos siempre el mismo proceso los humanos? La respuesta es que quizás no, porque quizás no aplicamos la misma técnica a ordenar dos palabras que a ordenar 200. Si tenemos una lista de 2 palabras, quizás analizamos cuál es la primera y asignamos el orden segundo a la que queda. En cambio, con una lista de 200, podemos analizar cuál es la primera... y empezar a tener problemas (por ejemplo, 24 palabras que tengan las primeras 3 letras iniciales iguales). A partir de ahí, empezamos a utilizar técnicas como el marcado de palabras y letras (mental o sobre el papel), asignación de órdenes relativos entre palabras empatadas y asignación de órdenes absolutos dentro de la lista, eliminación de palabras con orden determinado, etc.

Bien, "creemos" que haya algún algoritmo que nos permita ordenar cualquier lista de números o palabras, puesto que los humanos lo hacemos. Vamos a tratar de centrar el problema con las pautas expuestas en el curso "Bases de la programación nivel I" de [aprenderaprogramar.com](http://aprenderaprogramar.com):

1. ¿Cuál es mi objetivo?
2. ¿Cuáles son los condicionantes que afectan al problema?
3. ¿Qué método o esquema de resolución voy a aplicar?
4. ¿Cuáles son los datos de partida?
5. ¿Qué resultados quiero obtener?

### **1. ¿Cuál es mi objetivo?**

Desarrollar un programa capaz de ordenar una serie de series de números.

## 2. *¿Cuáles son los condicionantes que afectan al problema?*

Entre los posibles condicionantes que se nos ocurren como reflexión inicial tendríamos:

- Cantidad de series.
- Longitud de las series: ¿las series tienen todas igual cantidad de números? ¿o unas tienen más números que otras?
- Los números ¿son naturales?, ¿son enteros?, ¿son enteros excluido el cero...?
- Existencia de empates y su localización: ¿Existen empates en el primer número de la serie? ¿Sólo en el primer número y no en los siguientes? ¿No existen empates? ¿Existen pocos empates? ¿Existen muchos empates?

El planteamiento inicial con el que vamos a trabajar es el de orientación "genérica" pero a partir de:

- Cualquier número de series.
- Series de igual longitud.
- Números enteros positivos.
- Cualquier posibilidad de casos en relación a empates: inexistentes, escasos o abundantes.

## 3. *¿Qué método o esquema de resolución voy a aplicar?*

Desarrollaremos este punto de forma extensa posteriormente.

## 4. *¿Cuáles son los datos de partida?*

Dado que se trata de un programa de autodesarrollo o programa-herramienta para su uso en otros programas, vamos a plantear la siguiente estrategia:

- El número de filas y de columnas serán establecidos por el programador en el propio código.
- Los datos serán generados de forma aleatoria (por supuesto, controlada) o bien serán definidos por el programador a través de código del programa.

## 5. *¿Qué resultados quiero obtener?*

Nos vamos a limitar a obtener una pantalla donde el ordenador nos dice cuál es la matriz a ordenar (cuya generación controlaremos a través del código) y cuál es la matriz ordenada (resultado del algoritmo de ordenación que planteemos).

## **MÉTODO O ESQUEMA DE RESOLUCIÓN**

### **Documentarnos**

¿Cómo enfrentarnos al problema? Quizás hay distintas vías de resolución (igual que hay distintas formas para ordenar una serie de números) pero el primer paso, y será muy difícil empezar a hacer algo sin darlo, será el de documentarnos y plantear casos de series de números a ordenar para tratar de determinar una metodología común para resolverlos todos ellos. ¿Qué metodología de resolución usaremos? Podríamos pensar en ordenar los elementos de cada columna. Sin embargo, viendo unos

cuantos casos podemos llegar a la conclusión de que la matriz ordenada por columnas no nos permite resolver el problema, porque nos dice cuál es el mayor valor de cada columna pero no a qué serie pertenece ese valor ni si ese valor tiene importancia para determinar el orden de la serie dentro de la lista. Por tanto no basta con ordenar series de números.

Parece claro que vamos a necesitar "marcadores", o asociar a un valor por ejemplo la serie a que pertenece. ¿Cuántos marcadores utilizar? Seguramente se nos pueden ocurrir distintas alternativas. No obstante, casi el problema se restringe a dos opciones:

- a) Desarrollar un algoritmo de ordenación.
- b) Tratar de emular el procedimiento humano o "algoritmo humano".

Si buscáramos desarrollar un algoritmo "óptimo" podríamos optar por buscar información sobre métodos de ordenación de este tipo en libros, internet, revistas, etc. En nuestro caso el objetivo es didáctico y nos vamos a decantar por emular el proceder humano. Los motivos para ello:

1. Sabemos que funciona
2. Sabemos aplicarlo, aunque no tengamos "el código".
3. Probablemente nos resulte más sencillo trabajar partiendo de algo que ya sabemos que tratar de determinar nuevos métodos.
4. Todo ello nos lleva a que sea didáctico, uno de los fines de este curso.

Bien, vamos a emular el proceder humano y, ¿cómo lo hacemos?

A cada elemento de la matriz le podemos asignar un valor, el orden que ocupa dentro de la columna a la que pertenece y el orden de su serie dentro de la lista.

Elemento "origen"	Conceptos asociados
Elemento de la matriz que queremos ordenar	Valor numérico Orden dentro de su columna Orden de su serie

Una forma de plasmar este planteamiento con Visual Basic se basa en crear un tipo de datos que nos permita mantener toda esta información agrupada. En las versiones menos recientes de Visual Basic esto se hace con lo que se denomina tipos definidos por el usuario o variables UDT (User Defined Type), y en las versiones más recientes se denomina estructuras o Structures. No hemos estudiado UDT ni Structures en el curso pero vamos a usar este tipo de datos en este ejercicio porque resultan de utilidad, lo cual no significa que sean "imprescindibles". La creación de este tipo de datos se hace así:

#### Versiones menos recientes de VB:

```
Private Type element
    Valor As Single
    Ordenensucolumna As Integer
    Ordendesuserie As Integer
End Type
```

#### Versiones más recientes de VB:

```
Private Structure element
    Dim Valor As Single
    Dim Ordenensucolumna As Integer
    Dim Ordendesuserie As Integer
End Structure
```

El proceso humano, aunque puede tener sus variantes, sería algo similar a lo siguiente:

1. Análisis de columna y establecer órdenes relativos y empates.
2. Si se desprende un orden definitivo, asignarlo, eliminar la serie del proceso de evaluación y volver a comenzar el proceso de evaluación para todos los elementos restantes en la primera columna (esto es discutible pero lo haremos así de momento).
3. **En las palabras empatadas (sólo las empatadas)**, aplicar el punto 1 en la siguiente columna. Si persiste un empate, repetir el punto 3. Si se termina el proceso y persiste un empate es que dos o más series son iguales. En este caso, se les daría un número de orden indistinto, ya que son iguales, y se eliminarían del proceso de evaluación. Se continuaría volviendo a comenzar el proceso de evaluación para todos los elementos restantes en la primera columna.

En la siguiente entrega iremos viendo cómo terminar de plantear el problema y transformarlo en código.

**Próxima entrega: CU00357A**

**Acceso al curso completo en [aprenderaprogramar.com](http://www.aprenderaprogramar.com)** -- > Cursos, o en la dirección siguiente:  
[http://www.aprenderaprogramar.com/index.php?option=com\\_content&view=category&id=37&Itemid=61](http://www.aprenderaprogramar.com/index.php?option=com_content&view=category&id=37&Itemid=61)