

## **Arreglos**

Favian Orduña Suárez

Tecnológico Nacional de México Campus Querétaro

Fundamentos de programación

María Luisa Montes Almanza

14 de noviembre del 2025

## **Definición y propiedades de matrices**

Una matriz se concibe en el contexto de la programación como un arreglo bidimensional. Es una de las estructuras de datos más comunes y esenciales para organizar información de manera tabular.

### **Definición y Características Fundamentales**

Una matriz, al igual que cualquier arreglo, se define como una colección finita, homogénea y ordenada de datos.

1. Finita: Esto significa que el número total de elementos que la matriz puede contener está limitado y debe ser especificado. En una matriz, el tamaño está dado por la multiplicación del número de filas por el número de columnas ( $N \times M$ ).
2. Homogénea: Es imperativo que todos los elementos que componen la matriz sean del mismo tipo de dato (por ejemplo, todos deben ser enteros, o todos de punto flotante, o todos caracteres).
3. Ordenada: Es posible determinar claramente la posición de cada elemento dentro de la colección.

### **Acceso y Referencia**

La clave para manipular una matriz es su sistema de indexación. Para hacer referencia a cualquier componente individual, se necesitan dos índices: el primero indica la fila y el segundo indica la columna. Por ejemplo, para acceder a una calificación en la posición  $i, j$ , se usaría la notación  $A[i][j]$ .

## **Arreglos unidimensionales**

Los arreglos unidimensionales, a menudo denominados vectores, son las estructuras fundamentales para almacenar conjuntos de datos que comparten las mismas características.

### **Conceptos Básicos**

Un arreglo unidimensional es, por definición, una colección finita, homogénea y ordenada de datos.

- **Componentes e Índice:** Consta de dos partes esenciales: los componentes (los valores almacenados) y el índice (un número único que especifica la ubicación o casilla del elemento).
- **Declaración y Uso:** Cuando se declara un arreglo (por ejemplo, en un lenguaje como C), se reserva el espacio de memoria para contener esa cantidad de elementos. La inicialización puede darse al momento de la declaración o mediante ciclos de lectura. Además, es importante ver que en lenguajes como C, los arreglos se comportan de manera similar a los apuntadores, ya que el nombre del arreglo actúa como un apuntador hacia el primer elemento.

## **Operaciones y Aplicaciones**

Las operaciones típicas en arreglos unidimensionales se centran en la localización y reordenamiento de datos:

1. **Búsqueda Secuencial:** Consiste en examinar el arreglo elemento por elemento, generalmente de izquierda a derecha, hasta que se encuentra el dato deseado o se llega al final.
2. **Búsqueda Binaria:** Este es un método mucho más rápido, pero requiere que el arreglo esté previamente ordenado. Su eficacia radica en que, en cada iteración, compara el valor buscado con el elemento central del intervalo, reduciendo el espacio de búsqueda a la mitad.
3. **Ordenación por Selección Directa:** Un algoritmo de ordenamiento que funciona localizando el elemento más pequeño del arreglo no ordenado y lo coloca en la posición que le corresponde, repitiendo este paso hasta que el arreglo está completamente ordenado.

## **Aplicaciones Comunes:**

Los arreglos unidimensionales son ideales para:

- Almacenar y procesar listas de valores, como los sueldos de 70 empleados.
- Realizar cálculos estadísticos complejos, como la media aritmética, la varianza, la desviación estándar y la determinación de la moda dentro de un grupo de calificaciones.
- Contar y acumular información, como el número de votos obtenidos por cada candidato en una elección.

## **Multidimensionales**

El concepto de arreglo se expande a múltiples dimensiones para manejar problemas que requieren una indexación más compleja que la simple fila y columna.

### **Conceptos Básicos**

Un arreglo multidimensional se emplea cuando la referencia a los elementos necesita la utilización de múltiples índices. La cantidad de dimensiones necesarias depende directamente de las características del problema que se esté modelando.

Aunque el arreglo bidimensional (la matriz, que requiere dos índices) es el más común, existen arreglos de más de dos dimensiones, como los tridimensionales.

- Manejo de Datos: El manejo de estos arreglos, incluyendo la lectura, la escritura y el procesamiento, requiere el uso de ciclos anidados; se necesita un ciclo for (o similar) por cada dimensión. Por ejemplo, un arreglo tridimensional requerirá tres ciclos anidados para ser recorrido completamente.

### **Operaciones y Aplicaciones**

Los arreglos multidimensionales son cruciales para modelar escenarios del mundo real con variables interconectadas:

1. Operaciones Tabulares Complejas: Permiten realizar sumas de matrices o calcular la traspuesta.

2. Organización de Datos Jerárquicos: Se utilizan para organizar datos basados en múltiples criterios, como:

2.1. Registro Universitario (3D): Almacenar el número de alumnos, clasificándolos por carrera, semestre y año.

2.2. Ventas y Producción (3D): Registrar ventas mensuales por departamento a lo largo de varios años, o la producción mensual de un producto a lo largo de varios años.

2.3. Meteorología (3D): Almacenar promedios de temperatura o lluvia, clasificados por estado (o provincia), mes y año.

### **Estructuras o registros**

Las estructuras (o registros) son uno de los tipos de datos estructurados más importantes, caracterizados por una propiedad que los distingue fundamentalmente de los arreglos.

#### **Definición y Propiedades Clave**

Una estructura se define como una colección de elementos finita y, crucialmente, heterogénea.

- Heterogeneidad: Es la principal diferencia con los arreglos. Mientras que un arreglo exige que todos sus elementos sean del mismo tipo, una estructura permite almacenar campos de diferentes tipos de datos (por ejemplo, un campo de tipo entero para la matrícula, un campo de cadena para el nombre y un campo real para el promedio).
- Campos (Fields): Cada elemento constituyente de una estructura se conoce como campo (o field).

#### **Acceso y Aplicaciones Avanzadas**

El acceso a los datos dentro de un registro se realiza utilizando el nombre de la variable tipo estructura seguido del nombre del campo específico.

1. Estructuras Anidadas: Las estructuras pueden contener otras estructuras como uno de sus campos. Esto permite modelar datos complejos y jerárquicos, como el registro de un paciente que incluye un campo Domicilio, el cual es a su vez otra estructura con campos como Calle, Número, y Colonia.

2. Registros con Arreglos: Es muy común que un campo dentro de una estructura sea un arreglo. Por ejemplo, una estructura PROFESOR puede incluir un campo SALARIO que es un arreglo unidimensional para registrar los salarios mensuales a lo largo de un año. También pueden existir arreglos de registros (vectores de estructuras), donde cada elemento del vector es un registro completo (como una lista de ALUMNOS, donde cada alumno es una estructura).

3. Uniones (Unions): Conceptualmente cercanas a las estructuras, las uniones se diferencian en que sus miembros comparten la misma ubicación de almacenamiento en memoria. Esto significa que, si bien una unión puede tener varios campos, solo uno de ellos puede tener un valor asignado y ser accesible en un momento dado, optimizando así el uso de la memoria.

## **Referencias**

- Osvaldo Cairó Battistutti (2005). Metodología de la programación Algoritmos, diagramas de flujo programas (3a ed.). Editorial Alfaomega
- Corona Nakamura, María de los ángeles (2011). Diseño de algoritmos y su codificación en lenguaje c (1ª ed.). Editorial Mc Graw Hill
- Osvaldo Cairó Battistutti, O. (2006). *Fundamentos de programación. Piensa en C*. Pearson Educación.