

# Lenguaje de Programación

C++  
Arreglos

José Luis Alonzo Velázquez

Universidad de Guanajuato

Sesión 31

## Un arreglo en C++

Un **arreglo** es una colección ordenada de variables del mismo tipo. Las variables que pertenecen a un arreglo se conocen por el nombre de **elementos**.

El **término ordenado** significa que en la colección hay un **primer elemento**, un **segundo elemento**, un **tercer elemento**, y así sucesivamente.

Además, los elementos pueden a su vez organizarse en subgrupos llamadas **dimensiones**.

## Dimensiones

El subgrupo más pequeño posible se conoce como un arreglo de una dimensión. Un arreglo de dos dimensiones se subdivide en arreglos de una dimensión. Un arreglo de tres dimensiones se subdivide en arreglos de dos dimensiones los cuales a su vez se dividen en arreglos de una dimensión. Un arreglo de cuatro dimensiones se subdivide en arreglos de tres dimensiones los cuales a su vez se dividen en arreglos de dos dimensiones los cuales a su vez se dividen en arreglos de una dimensión. La misma idea se aplica en arreglos de más dimensiones. Más adelante veremos esto mas a detalle...

## Sintaxis

```
<tipo> nombre_variable [longitud];
```

Con esto diremos que `nombre_variable` es un arreglo de longitud `longitud` elementos del tipo `<tipo>`. Cabe destacar que `longitud` debe ser cualquier expresión entera constante mayor que cero.

## Sintaxis

```
<tipo> nombre_variable [longitud];
```

Con esto diremos que `nombre_variable` es un arreglo de longitud `longitud` elementos del tipo `<tipo>`. Cabe destacar que `longitud` debe ser cualquier expresión entera constante mayor que cero.

## Asignación de un arreglo

```
nombre_variable [índice] = expresión del tipo <tipo>
```

Esta instrucción asigna el valor asociado de la expresión a la posición `índice` del arreglo `nombre_variable`. El `índice` debe ser una expresión del tipo entero en el rango `[0, longitud-1]`. Cabe destacar que C++ no chequea que el valor de la expresión sea menor a `longitud`, simplemente asigna el valor a esa posición de memoria como si formara parte del arreglo, pisando, de esta manera, otros datos que no forman parte del mismo, con lo que finalmente el programa no funciona correctamente.

## En resumen, un arreglo:

- No es una variable; es un grupo de variables conocidas como elementos

## En resumen, un arreglo:

- No es una variable; es un grupo de variables conocidas como elementos
- Cada elemento ocupa una posición dentro del grupo

## En resumen, un arreglo:

- No es una variable; es un grupo de variables conocidas como elementos
- Cada elemento ocupa una posición dentro del grupo
- Todos los elementos son del mismo tipo

## En resumen, un arreglo:

- No es una variable; es un grupo de variables conocidas como elementos
- Cada elemento ocupa una posición dentro del grupo
- Todos los elementos son del mismo tipo
- El nombre del arreglo indica donde se localiza el grupo en la memoria de la computadora

## En resumen, un arreglo:

- No es una variable; es un grupo de variables conocidas como elementos
- Cada elemento ocupa una posición dentro del grupo
- Todos los elementos son del mismo tipo
- El nombre del arreglo indica donde se localiza el grupo en la memoria de la computadora
- Los arreglos se clasifican de acuerdo a las dimensiones que tengan

## En resumen, un arreglo:

- No es una variable; es un grupo de variables conocidas como elementos
- Cada elemento ocupa una posición dentro del grupo
- Todos los elementos son del mismo tipo
- El nombre del arreglo indica donde se localiza el grupo en la memoria de la computadora
- Los arreglos se clasifican de acuerdo a las dimensiones que tengan
- Las dimensiones no tienen relación con el plano Cartesiano; nada que ver con matemática

## En resumen, un arreglo:

- No es una variable; es un grupo de variables conocidas como elementos
- Cada elemento ocupa una posición dentro del grupo
- Todos los elementos son del mismo tipo
- El nombre del arreglo indica donde se localiza el grupo en la memoria de la computadora
- Los arreglos se clasifican de acuerdo a las dimensiones que tengan
- Las dimensiones no tienen relación con el plano Cartesiano; nada que ver con matemática
- Las dimensiones indican como están organizados los elementos dentro del grupo

## En resumen, un arreglo:

- No es una variable; es un grupo de variables conocidas como elementos
- Cada elemento ocupa una posición dentro del grupo
- Todos los elementos son del mismo tipo
- El nombre del arreglo indica donde se localiza el grupo en la memoria de la computadora
- Los arreglos se clasifican de acuerdo a las dimensiones que tengan
- Las dimensiones no tienen relación con el plano Cartesiano; nada que ver con matemática
- Las dimensiones indican como están organizados los elementos dentro del grupo
- Los arreglos de dos dimensiones pueden visualizarse como tablas

## En resumen, un arreglo:

- No es una variable; es un grupo de variables conocidas como elementos
- Cada elemento ocupa una posición dentro del grupo
- Todos los elementos son del mismo tipo
- El nombre del arreglo indica donde se localiza el grupo en la memoria de la computadora
- Los arreglos se clasifican de acuerdo a las dimensiones que tengan
- Las dimensiones no tienen relación con el plano Cartesiano; nada que ver con matemática
- Las dimensiones indican como están organizados los elementos dentro del grupo
- Los arreglos de dos dimensiones pueden visualizarse como tablas
- Los valores que se guarden en el arreglo se almacenan en los elementos ya que los elementos son las variables

Para crear arreglos en C++, hay que indicar:

- 1 el tipo de los elementos (ejemplo, int, char, double, bool o un tipo definido por el programador)
- 2 el nombre del arreglo
- 3 la cantidad de dimensiones y sus tamaños; cada dimensión comienza con el signo [ seguido por el tamaño de la dimensión y termina con el signo ]

Para crear un arreglo de una dimensión, el formato es el siguiente:

```
<tipo de los elementos> <nombre del arreglo> [ <tamaño primera dimensión> ]
```

Para crear arreglos en C++, hay que indicar:

- 1 el tipo de los elementos (ejemplo, int, char, double, bool o un tipo definido por el programador)
- 2 el nombre del arreglo
- 3 la cantidad de dimensiones y sus tamaños; cada dimensión comienza con el signo [ seguido por el tamaño de la dimensión y termina con el signo ]

Para crear un arreglo de una dimensión, el formato es el siguiente:

```
<tipo de los elementos> <nombre del arreglo> [ <tamaño primera dimensión> ]
```

### Cantidad de elementos en un arreglo

Para determinar la cantidad de elementos en un arreglo, hay que multiplicar el tamaño de cada una de las dimensiones indicados en la declaración. El tamaño debe ser un literal de tipo entero o el nombre de una constante de tipo entero.

## Observaciones

Para nombrar un elemento en un arreglo hay que indicar el nombre del arreglo, seguido de tantas parejas de corchetes [ ] como dimensiones declaradas. Dentro de cada pareja de corchetes tiene que haber un índice. La combinación de los índices indica la posición del elemento dentro del grupo. El valor mínimo para un índice es 0. El valor máximo es uno menos que el tamaño de la dimensión correspondiente en la declaración del arreglo.

## Ejemplo #1 Arreglo de una dimensión

Declaración

```
int a[3]; // forma una secuencia de tres elementos
```

Nombre del grupo

a

Nombre de los elementos

a[0] -> primer elemento

a[1] -> segundo elemento

a[2] -> tercer elemento

## Ejemplo #2 Arreglo de dos dimensión

Declaración

```
char m[2][3]; // forma una tabla de dos filas y tres columnas
// cada fila es un arreglo de una dimensión
// la declaración indica que hay dos arreglos de una dimensión
```

Nombre del grupo

```
m // indica la localización del grupo en la memoria
```

Nombre de las filas

```
m[0] // primera fila -> indica la localización de la fila dentro del grupo
m[1] // segunda fila -> indica la localización de la fila dentro del grupo
```

Nombre de los elementos

```
m[0][0] // primer elemento
m[0][1] // segundo elemento
m[0][2] // tercer elemento
m[1][0] // cuarto elemento
m[1][1] // quinto elemento
m[1][2] // sexto elemento
```

## Ejemplo #3 Arreglo de tres dimensión

### Declaración

```
double w[2][2][3]; // forma dos tablas de dos filas y tres columnas
// cada fila es una secuencia de tres elementos
// cada fila es un arreglo de una dimensión
// la declaración indica que hay dos arreglos de dos dimensiones
// los arreglos de dos dimensiones tienen a su vez dos arreglos de una dimensión
```

### Nombre del grupo

```
w // indica la localización del grupo en la memoria
```

### Nombre de las tablas

```
w[0] // primera tabla // indica la localización de la tabla dentro del grupo
w[1] // segunda tabla. indica la localización de la tabla dentro del grupo
```

### Nombre de las filas

```
w[0][0] // primera tabla, primera fila
w[0][1] // primera tabla, segunda fila
w[1][0] // segunda tabla, primera fila
w[1][1] // segunda tabla, segunda fila
```

### Nombre de los elementos

```
w[0][0][0] // primer elemento
w[0][0][1] // segundo elemento
w[0][0][2] // tercer elemento
w[0][1][0] // cuarto elemento
w[0][1][1] // quinto elemento
w[0][1][2] // sexto elemento
w[1][0][0] // séptimo elemento
w[1][0][1] // octavo elemento
w[1][0][2] // noveno elemento
w[1][1][0] // décimo elemento
w[1][1][1] // undécimo elemento
w[1][1][2] // duodécimo elemento
```

## Problema para clase

Crear un arreglo con 5 elementos e imprimir en pantalla todos los elementos del arreglo.

-  Como Programar en C/C++, Deitel (Prentice Hall), 2da Edición.
-  Programming Principles and Practice Using C++, Bjarne Stroustrup.
-  <http://www.codeblocks.org>
-  <http://www.wxwidgets.org>
-  (O'Reilly) Practical C Programming (3rd Edition)