

# ICF 122 - Fundamentos de Programación

## Ingeniería Civil Informática

### Unidad 5: Arreglos

Facultad de Ingeniería



**Universidad  
Andrés Bello**

- El **arreglo** es un tipo estructurado de dato, el cual es capaz de almacenar una colección de datos del mismo tipo.
- Es la estructura de datos más utilizada por los programadores.
- Es la forma más simple de agrupar **componentes de un mismo tipo** y asociarles un **número de orden** de cada componente llamado **índice**.
- Los arreglos en C, se almacenan en posiciones contiguas de memoria y poseen tamaño.

- El arreglo es una estructura multidimensional.
- En una dimensión puede ser visto como un **vector**, necesita de **un** índice (la posición) para recorrer sus elementos.
- Un arreglo de dos dimensiones puede representarse como una **matriz**, necesita de **dos** índices (fila y columna) para acceder a uno de sus elementos.
- En general un arreglo puede ser n-dimensional, y requerirá de n índices para recorrer o acceder a sus elementos.

# Arreglos

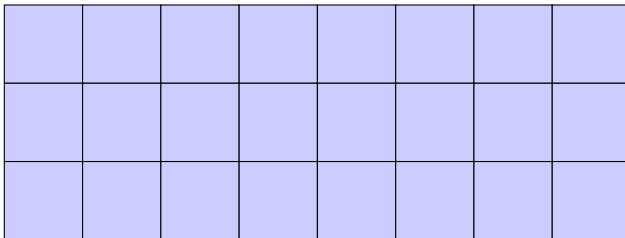
Unidimensional

```
int a[10];
```



Bidimensional

```
int a[3][8];
```



# Arreglos

```
int numeros[]={1,3,34,54}; //su tamaño es 4

//arreglo de caracteres de tamaño 5
char alfabeto[5]={'A','B','C','D','E'};

//3 filas y columnas a lo mas de 40 caracteres.
char nombres[][40]={"pedro", "pablo", "luis"};

//arreglo bidimensional de enteros
int coordenadas[2][2]={{0,0},{1,1}};
```

- Los índices de los arreglos comienzan en cero, es decir, el primer elemento, es el elemento cero.
- Si un arreglo es declarado en la función principal `main()` tendrá las dimensiones incluidas.
- Cuando se pasa un arreglo a una función, esta última no necesita saber el tamaño.

# Arreglos

## Unidimensionales

- Es una lista o vector.
- Declaración

```
tipo_dato nombreArreglo[tamano];
```

- `tipo_dato`: de qué tipo serán los elementos. Recordar que todos son del mismo tipo.
- `nombreArreglo`: nombre para la variable.
- `[]`: nos indica la dimensionalidad.
- `tamano`: cantidad de elementos que como máximo podrá contener el arreglo.

# Arreglos

## Unidimensionales

- Un elemento individual dentro de un arreglo es accedido por el uso de un **índice**.
- Un índice describe la posición de un elemento dentro de un arreglo.
- En C el primer elemento tiene el índice **cero**!

```
int v[5];
```

12	4	34	78	9
v[0]	v[1]	v[2]	v[3]	v[4]



# Arreglos

## Unidimensionales

```
#include <stdio.h>

int main(){
    int muestra[10],t;

    for(t = 0; t < 10; t++){
        muestra[t] = t*t;

    for(t = 0; t < 10; t++){
        printf("muestra[%d] = %d",t,muestra[t]);

    return 0;
}
```

- En C, un arreglo es mapeado a localizaciones de memoria continua.
- Todos los elementos de memoria residen uno al lado del otro.
- La dirección más baja corresponde al primer elemento, y la más alta al último elemento.
- La cantidad de bytes del arreglo de tamaño `n` está dada por `sizeof(tipo_dato)*n`.

# Arreglos

- No se puede asignar un arreglo a otro.
- Lo siguiente es ilegal:

```
int a[10], b[10];  
  
a = b; //esto es ilegal
```

- En vez, se debe hacer asignaciones por cada elemento:

```
int i;  
for(i = 0; i < 10; i++)  
    a[i] = b[i];
```

## Ejercicio 1

*Escribir un programa que permita solicitar 10 elementos y muestre por pantalla la media aritmética y la varianza de los datos. Supongamos que el  $i$ -ésimo dato viene dado por  $x_i$ , para  $i = 1, \dots, n$ , entonces la media aritmética viene dada por:*

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$$

*y la varianza viene dada por:*

$$var = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2$$