



Facultad de Ingeniería Ingeniería y de Sistemas

# Algoritmia y Estructura de Datos: Arreglos multidimensionales

# Arreglos bidimensionales

Identificador de la  
variable arreglo

columns  $j$

x

	0	1	2	3	4	5
0	14	8	19	16	7	12
1	11	15	18	13	12	9
2	6	17	22	4	1	21
3	19	9	31	17	25	3

filas  $i$

$x[2][3]$

Identificador de un  
elemento del arreglo

Estructura de filas y columnas de  
datos de un mismo tipo

# Asignación, lectura y escritura de elementos sucesivos de un arreglo bidimensional en C++

Cuando se requiere asignar, leer o escribir elementos sucesivos de un arreglo se utilizan 2 bucles

declaración

```
int x[4][6], i, j;
```

lectura

```
for(i=0; i<4; i++)  
    for(j=0; j<6; j++)  
        cin>>x[i][j];
```

asignación

```
for(i=0; i<4; i++)  
    for(j=0; j<6; j++)  
        x[i][j] = 0;
```

escritura

```
for(i=0; i<4; i++)  
    for(j=0; j<6; j++)  
        cout<<x[i][j];
```

Pone en cero a los 24 elementos del arreglo x, desde x[0][0] hasta x[3][5]

Escribe los 24 valores del arreglo x, desde x[0][0] hasta x[3][5]

## Problema con arreglo de dos dimensiones

Desarrollar un algoritmo que determine si una matriz cuadrada de orden n es simétrica

		j	
		x	0 1 2 3
		0	14 11 6 19
		1	11 15 17 9
		2	6 17 22 31
		3	19 9 31 17

$$n = 4$$

Es simétrica si para todo  $i, j$   $x[i][j] = x[j][i]$

Una matriz cuadrada es aquella que tiene igual número de filas y columnas, donde n es el número de filas y columnas

## Reto sobre el problema anterior

Desarrollar un algoritmo que determine si una matriz cuadrada de orden  $n$  es simétrica, sin recorrer todos los elementos de la matriz y terminar en el momento en que ya se sabe que no es simétrica

		$j$				
		$x$	0	1	2	3
$i$		0	14	11	6	19
$i$	1	11	15	17	9	
	2	6	18	22	31	
3		19	9	31	17	

$$n = 4$$

Es simétrica si para todo  $i, j$   $x[i][j] = x[j][i]$

Solo es necesita recorrer los valores que estan debajo o encima de la diagonal principal

## Problema sobre arreglos de dos dimensiones

Desarrollar un algoritmo que lea una matriz cuadrada de orden  $n$  impar, donde  $n \geq 5$  y determine la suma de los elementos que figuran en sus dos diagonales principales

		$j$		
		0	1	2
$i$	0	14	11	6
	1	11	15	17
2	6	17	22	31
3	19	9	31	17
4	14	11	6	19

Encontrando la relación  $i + j$  para los elementos de la diagonal opuesta

$i$	$j$
0	4
1	3
2	2
3	1
4	0


$$i + j = n - 1$$

# Problema sobre arreglos de dos dimensiones

Una empresa comercializadora de instrumentos musicales tiene las ventas mensuales en dinero, por cada tipo de instrumentos que vende (Cuerda, Viento, Percusión y Electrónicos). Desarrollar un algoritmo que determine cuál fue el mes en que ocurrió la mayor venta total y cuál fue la línea de instrumentos que tuvo la menor venta annual.

		mes												
		0	1	2	3	4	5	6	7	8	9	10	11	12
Fila 0 total por mes		0	61970	70825	65889	65910						54234	60202	81676
Cuerda		1	220307	15000	18300	16400	13570	...	...	...	...	10120	13430	20340
Viento		2	210450	12310	15670	12460	18790	...	...	...	...	14830	12750	18430
Percusión		3	240679	16550	17210	18395	16427	...	...	...	...	13934	16479	22430
Electrónicos		4	236112	18110	19645	18634	17123	...	...	...	...	15350	17543	20476
Columna 0 Total annual por tipo de instrumento														

Se requiere acumular por fila y por columna, luego hallar el máximo de la fila 0 y el mínimo columna 0

# Problema sobre arreglos de dos dimensiones

```
#include <iostream>
using namespace std;
main(){
    int i,j;
    float ventas[5][13],max,min;
    for(j=1;j<13;j++)
        ventas[0][j]=0;
    for(i=1;i<5;i++)
        ventas[i][0]=0;
    for(i=1;i<5;i++)
        for(j=1;j<13;j++) {
            cout<<"Ingrese ventas["<<i<<""]["<<j<<"]=";
            cin>>ventas[i][j];
            ventas[0][j]=ventas[0][j]+ventas[i][j];
            ventas[i][0]=ventas[i][0]+ventas[i][j];
        }
}
```

```
    max=ventas[0][1];
    for(j=2;j<13;j++)
        if (ventas[0][j]>max)
            max=ventas[0][j];
        cout<<"meses con mayor venta: ";
    for(j=1;j<13;j++)
        if (ventas[0][j]==max)
            cout<<j<<" ";
    min=ventas[1][0];
    for(i=2;i<5;i++)
        if (ventas[i][0]<min)
            min=ventas[i][0];
        cout<<"tipo de instrumento con menor venta anual: ";
    for(i=1;i<5;i++)
        if (ventas[i][0]==min)
            cout<<i<<" ";
}
```

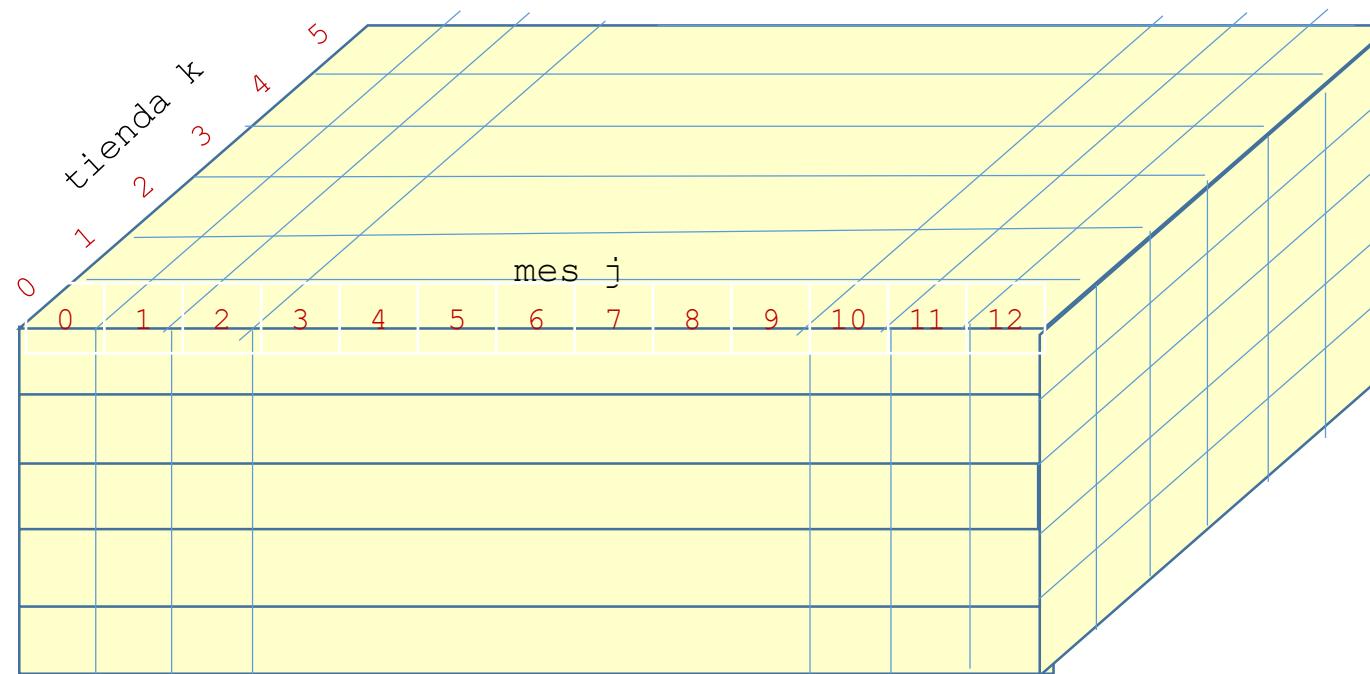
## Problema sobre arreglos de tres dimensiones

Una empresa comercializadora de instrumentos musicales tiene las ventas mensuales en dinero, por cada tipo de instrumentos que vende (Cuerda, Viento, Percusión y electrónicos) y cada una de las 5 tiendas con las que opera. Desarrollar un algoritmo que determine cuál fue el mes y tienda en que ocurrió la mayor venta total y cuál fue la línea de instrumentos y tienda que tuvo la menor venta anual.

```
float venta[5][13][6]
```

tipo instrumento i

venta



El total por mes y tienda se acumula en `venta[0][j][k]` y luego se halla el máximo.

El total anual por tipo de instrumento y tienda se acumula en `venta[i][0][k]` y luego se halla el mínimo