



- Practico de Laboratorio

# FUNCIONES ARREGLOS



Laboratorio Clase 7

Capítulo 11 – Sección 11.3 – Ejercicio 5.a)

*Escriba un programa en C++ que encuentre y despliegue el valor máximo en un arreglo bidimensional de números enteros. El arreglo deberá ser declarado como un arreglo de números enteros de 4 por 5 e inicializarse con los datos 16,22,99,4,18,  
-258,4,101,5,98,105,6,15,2,45,33,88,72,16,3.*

*Una vez encontrado el valor máximo se debe imprimir en pantalla además del número su subíndice de fila y columna donde se encontraba.*



## Laboratorio Clase 7

### Pseudocódigo:

*Declaro el arreglo y las variables*

*Inicializo el arreglo con los valores dados*

*Almaceno el primer valor como máximo.*

*Hago dos bucles for anidados para recorrer filas y columnas*

*Adentro del ciclo interior pregunto si el valor de la posición actual es mayor al máximo; en caso afirmativo guardo el valor y sus subíndice.*

*Imprimo los valores encontrados.*

### Prueba manual

**Verificación visual de cual es el máximo valor del arreglo**



### Codificación:

```
#include <iostream>
using namespace std;
```

```
int main( )
{ const int f=4, c=5;
```

Declaración e inicialización  
del arreglo bidimensional

```
int val[f][c] = {16,22,99,4,18,-258,4,101,5,98,105,6,15,2,45,33,88,72,16,3};
```

```
int i, j, subi, subj, mayor;
```

```
mayor=val[0][0];
```

```
for ( i=0 ; i<f ; i++ )
{
```

Bucle que recorre las filas

```
    for ( j=0 ; j<c ; j++ )
    {
```

Bucle que recorre las columnas

```
        if(val[i][j]>mayor)
        { mayor=val[i][j];
          subi=i;
          subj=j;
        }
```

```
    }
}
```

```
cout<<"El mayor valor del arreglo es : "<<mayor<<"de subindices"<<subi<<subj;
system("PAUSE");
return 0;
```



## Laboratorio Clase 7

### El código y la salida generada:

```
using namespace std;

int main()
{
    const int f=4, c=5;

    int val[f][c] = {16,22,99,4,18,-258,4,101,5,98,105,6,15,2,45,33,88,72,16,3};
    int i, j, subi, subj, mayor;

    mayor=val[0][0];
    for(i=0; i<f; i++)
    {cout<<endl;
        for(j=0; j<c; j++)
        {
            cout<<setw(4)<<val[i][j];
        }
    }

    for(i=0; i<f; i++)
        for(j=0; j<c; j++)
        {
            if(val[i][j]>mayor)
            {mayor=val[i][j];
                subi=i;
                subj=j;
            }
        }

    cout<<"\nEl mayor valor en el arreglo declarado "
        <<"se encontro en los subindices " <<subi<<" " <<subj
        <<" y el valor almacenado en el mismo es: " <<mayor<<endl;
```

```
16  22  99  4  18
-258  4 101  5  98
105  6  15  2  45
33  88  72 16  3
```

El mayor valor en el arreglo declarado se encontro en los subindices  
2 0 y el valor almacenado en el mismo es: 105

Presione una tecla para continuar . . .



## Laboratorio Clase 7

### **Paso 4: Prueba y corrección**

El objetivo de esta etapa es verificar que funciona en forma correcta y en realidad satisface los requerimientos.

Aquí verificamos la respuesta del programa, al calculo realizado en forma manual.

Se probó la ejecución del programa con los datos solicitados, y la verificación visual permitió corroborar los resultados. Podemos concluir que el programa cumple el algoritmo solicitado.



Laboratorio Clase 7

**Ejercicio 1.**

**Escriba un programa en C++ que permita introducir por teclado un arreglo bidimensional de números enteros de 2 filas por 3 columnas. Luego inicializar un segundo arreglo de números enteros de dos filas por 3 columnas con los siguientes valores:**

24	52	77
16	19	59

**Una vez ingresados los datos se deberán imprimir los arreglos en pantalla como tablas con una separación entre elementos de 4 caracteres.**

**Luego de mostrarlos por pantalla el programa deberá calcular el arreglo resultante de sumar cada elemento de ambos arreglos. Por ejemplo el elemento [1][2] del arreglo 1 mas el elemento [1][2] del arreglo 2 debe almacenarse en la posición [1][2] del arreglo resultante.**

**Finalmente se debera imprimir el arreglo resultante**



## Laboratorio Clase 7

Luego de la ejecución de este programa, y suponiendo que el usuario ingreso el valor 5. (n=5)

**Determinar la salida impresa en pantalla**

```
#include <iostream>
#include <iomanip>
using namespace std;
const int TAM = 100;
int main()
{
    int i, n, p = 0, A[TAM][TAM];
    cout << "Ingrese n: ";
    cin >> n;
    for (int i = 0; i < n; i++)
        for (int j = 0; j <= n; j++)
            A[i][j] = i;
    for (int i = 0; i < n; i = i + 2)
        for (int j = 1; j < n; j = j + 2)
            p += A[i][j];
    cout << p;
    cin.ignore(2);
    return 0;
}
```





Laboratorio Clase 7

**Ejercicio 2.**

***Escriba un programa en C++ que permita introducir un numero para el calculo de su potencia al cuadrado y/o su raíz cuadrada.***

***Una vez ingresado el numero solicitar el ingreso de un carácter: “r” para el calculo de raíz cuadrada, “c” para el calculo del cuadrado o “a” para el calculo de ambas funciones matemáticas (raíz y potencia)***

***Escriba una función llamada calculos() que calcule la/s funciones matemáticas de acuerdo al carácter ingresado. Esta función debe recibir como parámetro el numero y como segundo parámetro el carácter. La función deberá ser capaz de elevar al cuadrado números con puntos decimales.***

***Finalmente la función debe imprimir en pantalla el valor de la función matemática correspondiente según la elección realizada por el usuario; si se ingresa un carácter distinto a “r”, “c” o “a” se debe desplegar en pantalla un mensaje que indique que se ingreso un carácter invalido.***



## Laboratorio Clase 7

### Capítulo 6 – Sección 6.1 – Versión Ejercicio 3.a y b)

*Escriba un programa en C++ que solicite al usuario el ingreso de dos números para su posterior multiplicación.*

*Escriba una función llamada `mult()` que acepte estos dos números en punto flotante como parámetros, multiplique estos dos números y despliegue el resultado.*

*Incluya esta función escrita en un programa que funcione. Asegures de invocar su función desde `main`, y pruebe la función transmitiéndole valores varios.*