

Uso de arreglos de una dimensión como parámetro

Para declarar un arreglo como parámetro formal de una función se utiliza:

```
tipo nom_parametro[ ];
```

No es necesario especificar el tamaño del arreglo, sin embargo, al programar la función es necesario saberlo; por lo que es muy

común agregar un parámetro de tipo entero en el que se envía la cantidad de casillas ocupadas del arreglo.

Por default, un arreglo se considera un parámetro por referencia (de entrada y/o salida); es decir, cuando se recibe un arreglo

como parámetro y dentro de la función se modifica alguno de sus valores, el parámetro actual que se usó al mandar llamar a dicha

función se modifica; por otro lado, si se desea que el arreglo actúe como un parámetro por valor (de entrada); es decir, que no se

modifique dentro de la función se debe anteponer la palabra `const` a la declaración del parámetro; esto es en realidad una

precaución con la que se está avisando al compilador que esa función no debe modificar el valor de la variable; en caso de que por

error en la función se tratara de modificar el valor de alguna casilla del arreglo el compilador debe enviar un error indicándolo.

Un arreglo no puede ser regresado como valor de retorno de una función; por lo que si se requiere una función que regrese un

arreglo, éste se debe especificar como parámetro.

A continuación se encuentra un ejemplo en el que se incluye una función que recibe como parámetro un arreglo de precios, su

tamaño y un incremento, la función regresa el arreglo con los precios nuevos, es decir, después de aplicarles el incremento

indicado en el tercer parámetro:

```
#include <iostream.h>

void incrementa(double precios[], int tam, double inc)
{
    for (int i = 0; i < tam; i++)
        precios[i] = precios[i] * (1 + inc);
}

void lee(double arr[], int tam)
{
    for (int i = 0; i < tam; i++)
    {
        cout<<"Teclea el valor "<<i<<" ";
        cin>>arr[i];
    }
}

void muestra(const double arr[], int tam)
{
    for (int i = 0; i < tam; i++)
        cout<<"precio "<<i<<" "<<arr[i]<<endl;
}

int main()
{
    double precios[20];
    int tam;
    double incremento;

    cout<<"Cuantos precios hay en la lista?"<<endl;
    cin>>tam;
    cout<<"Teclea los precios "<<endl;
    lee(precios, tam);

    cout<<"cual es el incremento para los precios "<<endl;
    cout<<"teclea un valor entre 0 y 1 ";
    cin>>incremento;
    incrementa(precios, tam, incremento);

    // muestra la tabla de precios con el incremento
    muestra(precios, tam);

    return 0;
}
```

Uso de arreglos multidimensionales como parámetro

Cuando se requiere un arreglo de 2 o más dimensiones como parámetro no se requiere especificar el tamaño de la primera

dimensión; pero si se requiere el tamaño de las siguientes dimensiones. Como la primera dimensión no es necesaria, normalmente se

agrega un parámetro adicional de tipo entero que contiene la cantidad de datos de la primera dimensión del arreglo.

Revisemos el siguiente ejemplo en el que aparece como parámetro un arreglo de una dimensión y de dos dimensiones en la función

CopiaCol:

```
#include <iostream.h>

//Funcion que copia la columna A de la matriz al arreglo Arr
void CopiaCol(int Mat[][10], int orden, int A, int Arr[])
{
    for(int i=0; i<orden; i++)
        Arr[i] = Mat[i][A];
}

//Funcion que pide el orden de la matriz y luego lee sus elementos
void LeeMat(int Mat[][10], int &orden)
{
    cout<<"Cual es el orden de la matriz? ";
    cin>>orden;
    for (int ren=0; ren < orden; ren++)
        for (int col=0; col < orden; col++)
        {
            cout<<"Teclea el valor para el elemento "<<ren<<","<<col<<" ";
            cin>>Mat[ren][col];
        }
}

//Funcion que muestra los elementos de la matriz en la forma tradicional
void MuestraMat(int Mat[][10], int orden)
{
    for (int ren=0; ren<orden; ren++)
    {
        for (int col=0; col<orden; col++)
            cout<<Mat[ren][col]<<"\t";
        cout<<endl;
    }
}

//Funcion que muestra los elementos del arreglo
void MuestraArr(int Arr[], int orden)
{
    for (int ren=0; ren<orden; ren++)
        cout<<Arr[ren]<<"\t";
}

int main()
{
    int Matriz[10][10];
    int A, orden;
    int Arreglo[10];

    LeeMat(Matriz, orden);
    MuestraMat(Matriz, orden);
    cout<<"Copiar la columna A, Cual es el valor de A?";
    cin>>A;

    CopiaCol(Matriz, orden, A, Arreglo);
    cout<<"muestra el arreglo que tiene la columna "<<A<<" de la
matriz"<<endl;
    for (int i=0; i<orden; i++)
        cout<<Arreglo[i]<<"\t";
    cout<<endl;

    return 0;
}
```

```
}
```

Ejercicio

Realiza un programa que pida al usuario el orden (es decir, las dimensiones) de una matriz cuadrada, después que pida los valores

para las casillas del arreglo y después que muestre en la pantalla los valores que se encuentran en las 2 diagonales de la matriz.

[ver solución](#)

Ligas sugeridas

<http://www.cplusplus.com/doc/tutorial/>

<http://www.cs.wustl.edu/~schmidt/C++/>

[*Regresar*](#)

[*Siguiente módulo*](#)