



UNIVERSIDAD
TECNOLÓGICA
DE PANAMÁ

Facultad de Ingeniería de Sistemas Computacionales

Capítulo III

Arreglos

Herramienta de Programación Aplicada I

Kexy Rodríguez

Contenido

- Arreglos Multidimensionales.

Definición de Arreglos 2D

“Los arreglos vistos anteriormente se conocen como arreglos unidimensionales (una sola dimensión) y se caracterizan por tener un solo subíndice. Estos arreglos se conocen también por el término listas.

Los arreglos multidimensionales son aquellos que tienen más de una dimensión y, en consecuencia, más de un índice.

Los arreglos más usuales son los de dos dimensiones, conocidos también por el nombre de tablas o matrices, llamados Bidimensionales”

Definición

Arreglos

Es una secuencia de datos del mismo tipo. Se caracterizan por tener un solo índice o subíndice

Los datos se llaman elementos del arreglo y se numeran consecutivamente 0,1,2,3,etc

El tipo de elemento almacenado en el arreglo puede ser cualquier tipo de dato del lenguaje C

Tabla o Lista

Los elementos de un arreglo se numeran, estos números se denominan **índice ó subíndice**

El índice localizan la posición del elemento dentro del arreglo, proporcionando acceso directo al arreglo

Los arreglos en el lenguaje C se almacenan en celdas de memoria contiguas.

Estructura de un Arreglo de Dos Dimensiones

matriz	
índice	
	0 1 2 3 4 5 6 7 8
0	76 4 0 84 21 34 92 0 6
1	1 2 3 4 5 6 7 8 9
2	21 34 92 0 6 76 4 0 84

Sintaxis para Declarar una Matriz

tipo-base identificador[NumFilas][NumColum] ;

Observaciones:

El **tipo-base** puede ser cualquiera de los tipos básicos del lenguaje, o incluso algunos complejos como estructuras. Cada elemento de la matriz será del tipo definido.

El **identificador** es el nombre que distinguirá la matriz.

Los **corchetes** [] son obligatorios.

El término **NumFilas** determina el tamaño de la primera dimensión de la matriz, es decir, la cantidad de arreglos que existen en la primera dimensión. Los elementos (arreglos) de la segunda dimensión están numerados en forma consecutiva, empezando en 0. El término **NumColum** determina el tamaño de la segunda dimensión de la matriz, es decir, la cantidad de elementos del tipo base que contendrá. Dichos elementos estarán numerados en forma consecutiva, empezando en 0.

Almacenamiento en memoria de la matriz

Ejemplo:

```
int tabla[4][2];
```

Se representan gráficamente en memoria

```
tabla[0][0] =10;  
tabla[0][1] =20;  
tabla[1][0] =30;  
tabla[1][1] =40;  
tabla[2][0] =50;  
tabla[2][1] =60;  
tabla[3][0] =70;  
tabla[3][1] =80;
```

	0	1
0	10	20
1	30	40
2	50	60
3	70	80

Inicialización de una matriz

int tabla [2] [3] = {51,52,53,54,55,56};

```
#include <stdio.h>
int main() {
    //Bloque de declarativa
    int x, y, tabla[2][3] = {51,52,53,54,55,56};

    for (x=0;x<2;x++){
        for(y=0;y<3;y++){
            printf("%d ", tabla[x][y]);
        }
        printf("\n");
    }
    return 0;
}
```

	0	1	2
0	51	52	53
1	54	55	56

Ejemplo con cadena

```
#include <stdio.h>
int main() {
    //Bloque de declarativa
    char listaNombres[5][8];
    int x;
    //Recorre la matriz para insertar los nombres
    for(x=0;x<5;x++){
        printf("Introduzca el Nombre N. %d:",x+1);
        scanf("%s",&listaNombres[x]);
    }
    //Recorre la matriz para imprimir los nombres
    for(x=0;x<5;x++){
        printf("%s\n",listaNombres[x]);
    }
    return 0;
}
```

	0	1	2	3	4	5	6	7
0	J	u	a	n	a			
1	P	e	d	r	o			
2	M	a	r	i	a			
3	J	u	l	i	a	n	a	
4	L	u	i	s				

Tema para investigar

Investigue sobre la aplicación de las siguientes funciones:

- sizeof
- clrscr
- strcpy
- strlen
- strcat
- strcmp



UNIVERSIDAD
TECNOLÓGICA
DE PANAMÁ

